(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
31 March 2016 (31.03.2016)

WIPO | PCT

(10) International Publication Number
WO 2016/048784 A1

(51) International Patent Classification:
H04L 9/08 (2006.01)

(21) International Application Number:
PCT/US2015/050670

(22) International Filing Date:
17 September 2015 (17.09.2015)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
62/055,731 26 September 2014 (26.09.2014) US
62/098,391 31 December 2014 (31.12.2014) US

(71) Applicant: **THOMSON LICENSING** [FR/FR]; 1-5 rue Jeanne d'Arc, 92130 Issy les Moulineaux (FR).

(72) Inventor: **JOYE, Marc**; 183 El Carmelo Ave, Palo Alto, California 94306 (US).

(74) Agents: **SHEDD, Robert D.** et al.; 3rd floor, 4 Research Way, Princeton, New Jersey 08540 (US).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report (Art. 21(3))*

(54) Title: ANONYMOUS IDENTITY-BASED CRYPTOSYSTEMS

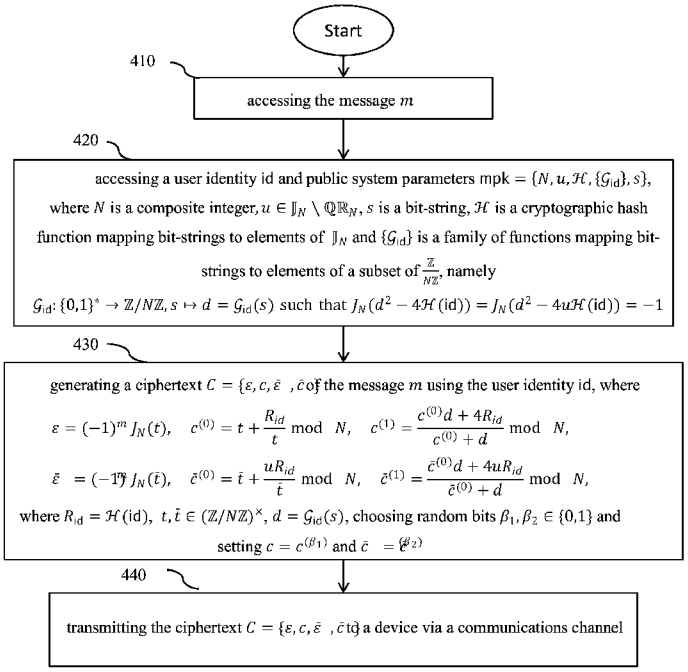

Figure 4

(57) Abstract: The present principles relate to several new identity-based cryptosystems. Notably, the cryptosystems are anonymous, that is, the ciphertexts reveal nothing about the recipient's identity. The ciphertexts are also twice shorter than the best known scheme. Further, the proposed cryptosystems come with strong security guarantees: they are proved to be semantically secure under the standard quadratic residuosity assumption, in the random oracle model. Anonymous identity-based schemes are important cryptographic tools as they constitute the central building block for public-key encryption with keyword search (a.k.a. PEKS).

WO 2016/048784 A1

ANONYMOUS IDENTITY-BASED CRYPTOSYSTEMS

RELATED APPLICATION

This patent application claims the benefit of U.S. Provisional Application No. 62/098391 filed on December 31, 2014, U.S. Provisional Application No. 62/055731 filed on September 26, 2014 both with the same title, and the disclosure of which is incorporated by reference herein in its entirety.

Field of the Invention

The present principles relate to cryptography, and more specifically, to anonymous identity-based encryption (IBE) systems.

Background Information

This section is intended to introduce the reader to various aspects of art, which may be related to various aspects of the present invention that are described and/or claimed below. This discussion is believed to be helpful in providing the reader with background information to facilitate a better understanding of the various aspects of the present invention. Accordingly, it should be understood that these statements are to be read in this light, and not as admissions of prior art.

Referenced Documents

[1] Michel Abdalla, Mihir Bellare, Dario Catalano, Eike Kiltz, Tadayoshi Kohno, Tanja Lange, John Malone-Lee, Gregory Neven, Pascal Paillier, and Haixia Shi. Searchable encryption revisited: Consistency properties, relation to anonymous IBE, and extensions. In V. Shoup, editor, *Advances in Cryptology – CRYPTO 2005*, volume 3621 of *Lecture Notes in Computer Science*, pages 205–222. Springer, 2005.

[2] Michel Abdalla, Mihir Bellare, Dario Catalano, Eike Kiltz, Tadayoshi Kohno, Tanja Lange, John Malone-Lee, Gregory Neven, Pascal Paillier, and Haixia Shi. Searchable encryption revisited: Consistency properties, relation to anonymous IBE, and extensions. *J. Cryptology*, 21(3):350–391, 2008. An extended abstract appears in [1].

[3] Giuseppe Ateniese and Paolo Gasti. Universally anonymous IBE based on the quadratic residuosity assumption. In M. Fischlin, editor, *Topics in Cryptology – CT-RSA 2009*, volume 5473 of *Lecture Notes in Computer Science*, pages 32–47.

Springer, 2009.

[4] Adam Barth, Dan Boneh, and Brent Waters. Privacy in encrypted content distribution using private broadcast encryption. In G. Di Crescenzo and A.D. Rubin, editors, *Financial Cryptography and Data Security*, volume 4107 of *Lecture Notes in Computer Science*, pages 52–64. Springer, 2006.

[5] Mihir Bellare, Alexandra Boldyreva, Anand Desai, and David Pointcheval. Key-privacy in public-key encryption. In C. Boyd, editor, *Advances in Cryptology – ASIACRYPT 2001*, volume 2248 of *Lecture Notes in Computer Science*, pages 566–582. Springer, 2001.

[6] Mihir Bellare and Phillip Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In *st ACM Conference on Computer and Communications Security*, pages 62–73. ACM Press, 1993.

[7] Dan Boneh, Giovanni Di Crescenzo, Rafail Ostrovsky, and Giuseppe Persiano. Public key encryption with keyword search. In C. Cachin and J. Camenisch, editors, *Advances in Cryptology – EUROCRYPT 2004*, volume 3027 of *Lecture Notes in Computer Science*, pages 506–522. Springer, 2004.

[8] Dan Boneh and Matthew K. Franklin. Identity-based encryption from the Weil pairing. *SIAM J. Comput.*, 32(3):586–615, 2003.

[9] Dan Boneh, Craig Gentry, and Michael Hamburg. Space-efficient identity based encryption without pairings. In *th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2007)*, pages 647–657. IEEE Computer Society, 2007.

[10] Michael Clear, Hitesh Tewari, and Ciarán McGoldrick. Anonymous IBE from quadratic residuosity with improved performance. In D. Pointcheval and D. Vergnaud, editors, *Progress in Cryptology – AFRICACRYPT 2014*, volume 8469 of *Lecture Notes in Computer Science*, pages 377–397. Springer, 2014.

[11] Giovanni Di Crescenzo and Vishal Saraswat. Public key encryption with searchable keywords based on Jacobi symbols. In K. Srinathan, C.P. Rangan, and M. Yung, editors, *Progress in Cryptology – INDOCRYPT 2007*, volume 4859 of *Lecture Notes in Computer Science*, pages 282–296. Springer, 2007.

[12] Shafi Goldwasser and Silvio Micali. Probabilistic encryption. *J. Comput. Syst. Sci.*, 28(2):270–299, 1984.

[13] Shai Halevi. A sufficient condition for key-privacy. IACR Cryptology ePrint Archive, Report 2005/005, 2005. http://eprint.iacr.org/.

[14] Marc Joye. U.S. Provisional Application No. 62/055722 filed on September 26, 2014, titled "*Key-private Cryptosystems Based on the Quadratic Residuosity.*"

[15] Marc Joye. U.S. Provisional Application No. 62/055729 filed on September 26, 2014, titled "*Identity-based XOR homomorphic cryptosystems.*"

[16] Marc Joye and Gregory Neven, editors. *Identity-Based Cryptography*, volume 2 of *Cryptology and Information Security Series*. IOS Press, 2009.

[17] Aggelos Kiayias, Yiannis Tsiounis, and Moti Yung. Group encryption. In K. Kurosawa, editor, *Advances in Cryptology − ASIACRYPT 2007*, volume 4833 of *Lecture Notes in Computer Science*, pages 181–199. Springer, 2007.

[18] Karl Rubin and Alice Silverberg. Compression in finite fields and torus-based cryptography. *SIAM J. Comput.*, 37(5):1401–1428, 2008.

[19] Adi Shamir. Identity-based cryptosystems and signature schemes. In G. R. Blakley and D. Chaum, editors, *Advances in Cryptology, Proceedings of CRYPTO '84*, volume 196 of *Lecture Notes in Computer Science*, pages 47–53. Springer, 1985.

Identity-based encryption

Identity-based cryptography is an extension of the public-key paradigm which was first put forward by Shamir [19]. A major issue with public-key cryptography is the management of trust. Another issue to be dealt with is to recover the public key and companion certificate, check them, and then only encrypt and send messages. We refer the reader to the excellent introduction [16] by Joux for details. Identity-based cryptography aims at solving these practical issues by simplifying the key management.

Formally, we define an *identity-based encryption scheme* [8] (or IBE in short) as a tuple of four algorithms $(\mathsf{SETUP}, \mathsf{EXTRACT}, \mathsf{ENCRYPT}, \mathsf{DECRYPT})$. The generalized flow diagram of the process is illustrated as process 300 in Figure 3.

Setup The setup algorithm SETUP 310 is a randomized algorithm that takes on input some security parameter κ and outputs the system parameters mpk together with the master secret key msk: $(\mathsf{mpk}, \mathsf{msk}) \xleftarrow{R} \mathrm{SETUP}(1^{\kappa})$.

Key derivation The key derivation algorithm EXTRACT , or Key Derivation 320, takes on input an identity id and master secret key msk and returns a secret key for the user with identity id: $\mathsf{usk} \leftarrow \mathrm{EXTRACT}_{\mathsf{msk}}(\mathsf{id})$.

Encryption Let $\mathcal{M}$ denote the message space. The encryption algorithm ENCRYPT 330 is a randomized algorithm that takes on input an identity id and a plaintext $m \in \mathcal{M}$, and returns a ciphertext C. We write $C \leftarrow \mathrm{ENCRYPT}_{\mathsf{mpk}}(\mathsf{id}, m)$.

Decryption The decryption algorithm DECRYPT 340 takes on input secret key usk (corresponding to identity id) and ciphertext C and returns the corresponding plaintext m or a special symbol $\perp$ indicating that the ciphertext is invalid. We write $m \leftarrow \mathrm{DECRYPT}_{\mathsf{usk}}(C)$ if C is a valid ciphertext and $\perp \leftarrow \mathrm{DECRYPT}_{\mathsf{usk}}(C)$ if it is not.

We require that, with non-negligible probability, $\mathrm{DECRYPT}_{\mathsf{usk}}(\mathrm{ENCRYPT}_{\mathsf{mpk}}(\mathsf{id}, m)) = m$ for all messages $m \in \mathcal{M}$. The specific processes associated with each of these steps as they relate to the present principles is described in further detail below.

Security notions

Indistinguishability (IND) of encryptions

The notion of *indistinguishability of encryptions* [12] captures a strong notion of data-privacy: The adversary should not learn any information whatsoever about a plaintext given its encryption beyond the length of the plaintext. The definitions for the public-key setting naturally extend to the identity-based paradigm. The standard definition is strengthened by allowing the adversary to issue chosen private-key extraction queries [8].

We view an adversary $\mathcal{A}$ as a pair $(\mathcal{A}_1, \mathcal{A}_2)$ of probabilistic algorithms. This corresponds to adversary $\mathcal{A}$ running in two stages. Upon receiving the system parameters mpk, in the "find" stage, algorithm $\mathcal{A}_1$ issues private-key extraction queries $\mathsf{id}_1, \ldots, \mathsf{id}_{n_1}$ and receives back the private key usk_i corresponding to identity id_i: $\mathsf{usk}_i \leftarrow \mathrm{EXTRACT}_{\mathsf{msk}}(\mathsf{id}_i)$. The queries may be asked adaptively. Once the adversary decides no to make further oracle queries, it outputs a challenge identity id^* (with $\mathsf{id}^* \neq \mathsf{id}_i$, $1 \leq i \leq n_1$), two (different) equal-size messages m_0 and $m_1 \in \mathcal{M}$, and some state information s. In the "guess" stage, algorithm $\mathcal{A}_2$ receives a challenge ciphertext C which is the encryption of m_b for identity id^* and where b is chosen at random in $\{0,1\}$. Algorithm $\mathcal{A}_2$ can issue more private-key extraction queries

$\mathsf{id}_{n_1+1}, \ldots, \mathsf{id}_{n_2}$; the only restriction is that $\mathsf{id}_i \neq \mathsf{id}^*$, $n_1 < i \leq n_2$. The goal of $\mathcal{A}_2$ is to recover the value of b from s and C.

An IBE scheme is said *semantically secure* (or *indistinguishable*) if

$$\Pr\left[\begin{array}{l}(\mathsf{mpk}, \mathsf{msk}) \overset{R}{\leftarrow} \mathrm{SETUP}(1^\kappa), \\ (\mathsf{id}^*, m_0, m_1, s) \leftarrow \mathcal{A}_1^{\mathrm{EXTRACT}_{\mathsf{msk}}(\cdot)}(\mathsf{mpk}), : \mathcal{A}_2^{\mathrm{EXTRACT}_{\mathsf{msk}}(\cdot)}(s, C) = b \\ b \overset{R}{\leftarrow} \{0,1\}, C \leftarrow \mathrm{ENCRYPT}_{\mathsf{mpk}}(\mathsf{id}^*, m_b)\end{array}\right] - \frac{1}{2}$$

is negligible in the security parameter for any polynomial-time adversary $\mathcal{A}$; the probability is taken over the random coins of the experiment according to the distribution induced by SETUP and over the random coins of the adversary.

Adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ can encrypt any message of its choice for any identity. In other words, the adversary can mount chosen-identity, chosen-plaintext attacks (ID-CPA). Hence, we write IND-ID-CPA the security notion achieved by a semantically secure identity-based encryption scheme.

Remark 1. As messages m_0 and m_1 are supposed to be different, when the message space is $\mathcal{M} = \{0,1\}$, the previous relation simplifies to

$$\Pr\left[\begin{array}{l}(\mathsf{mpk}, \mathsf{msk}) \overset{R}{\leftarrow} \mathrm{SETUP}(1^\kappa), \\ (\mathsf{id}^*, s) \leftarrow \mathcal{A}_1^{\mathrm{EXTRACT}_{\mathsf{msk}}(\cdot)}(\mathsf{mpk}), \quad : \mathcal{A}_2^{\mathrm{EXTRACT}_{\mathsf{msk}}(\cdot)}(s, C) = b \\ b \overset{R}{\leftarrow} \{0,1\}, C \leftarrow \mathrm{ENCRYPT}_{\mathsf{mpk}}(\mathsf{id}^*, b)\end{array}\right] - \frac{1}{2}.$$

Anonymity Analogously, the notion of *anonymity* captures a strong requirement about privacy: a ciphertext should not reveal the identity of the recipient. More formally, it is defined as a straightforward adaptation of key privacy [5] to the identity-based paradigm [2].

As before, we view an adversary $\mathcal{A}$ as a pair $(\mathcal{A}_1, \mathcal{A}_2)$ of probabilistic algorithms. In the "find" stage, algorithm $\mathcal{A}_1$ algorithm $\mathcal{A}_1$ issues private-key extraction queries $\mathsf{id}_1, \ldots, \mathsf{id}_{n_1}$ and receives back the private key usk_i corresponding to identity id_i: $\mathsf{usk}_i \leftarrow \mathrm{EXTRACT}_{\mathsf{msk}}(\mathsf{id}_i)$. The queries may be asked adaptively. Once the adversary decides not to make further oracle queries, it outputs two (different) challenge identities id_0^* and id_1^* (with $\mathsf{id}_0^*, \mathsf{id}_1^* \neq \mathsf{id}_i$, $1 \leq i \leq n_1$), a message $m \in \mathcal{M}$, and some state information s. In the "guess" stage, algorithm $\mathcal{A}_2$ receives a challenge ciphertext C which is the encryption of m for identity id_b^* and where b is chosen at

random in {0,1}. Algorithm $\mathcal{A}_2$ can issue more private-key extraction queries $\mathsf{id}_{n_1+1}, \ldots, \mathsf{id}_{n_2}$; the only restriction is that $\mathsf{id}_i \neq \mathsf{id}_0^*, \mathsf{id}_1^*$, $n_1 < i \leq n_2$. The goal of $\mathcal{A}_2$ is to recover the value of b from s and C.

An IBE scheme is said *anonymous (ANO)* if

$$\Pr\left[\begin{array}{l}(\mathsf{mpk},\mathsf{msk}) \overset{R}{\leftarrow} \mathrm{SETUP}(1^\kappa),\\ (\mathsf{id}_0^*, \mathsf{id}_1^*, m, s) \leftarrow \mathcal{A}_1^{\mathrm{EXTRACT}_{\mathsf{msk}}(\cdot)}(\mathsf{mpk}),: \mathcal{A}_2^{\mathrm{EXTRACT}_{\mathsf{msk}}(\cdot)}(s, C) = b\\ b \overset{R}{\leftarrow} \{0,1\}, C \leftarrow \mathrm{ENCRYPT}_{\mathsf{mpk}}(\mathsf{id}_b^*, m)\end{array}\right] - \frac{1}{2}$$

is negligible in the security parameter for any polynomial-time adversary $\mathcal{A}$; the probability is taken over the random coins of the experiment according to the distribution induced by SETUP and over the random coins of the adversary. We write ANO-ID-CPA the corresponding security notion achieved by an anonymous IBE.

Of course, the goals of indistinguishability and anonymity can be combined to give rise to the ANO-IND-ID-CPA security notion. Halevi's sufficient condition [13] was extended to IBE schemes in [2]. An IBE scheme is ANO-IND-ID-CPA if it is IND-ID-CPA and if

$$\Pr\left[\begin{array}{l}(\mathsf{mpk},\mathsf{msk}) \overset{R}{\leftarrow} \mathrm{SETUP}(1^\kappa),\\ (\mathsf{id}_0^*, \mathsf{id}_1^*, m, s) \leftarrow \mathcal{A}_1^{\mathrm{EXTRACT}_{\mathsf{msk}}(\cdot)}(\mathsf{mpk}),\\ b \overset{R}{\leftarrow} \{0,1\}, r \overset{R}{\leftarrow} \mathcal{M} \text{ and } |r|_2 = |m|_2,\\ \quad C \leftarrow \mathrm{ENCRYPT}_{\mathsf{mpk}}(\mathsf{id}_b^*, r)\end{array} : \mathcal{A}_2^{\mathrm{EXTRACT}_{\mathsf{msk}}(\cdot)}(s, C) = b\right] - \frac{1}{2}$$

is negligible in the security parameter for any polynomial-time adversary $\mathcal{A}$; the probability is taken over the random coins of the experiment according to the distribution induced by SETUP and over the random coins of the adversary. The difference is that a *random* message r is encrypted as opposed to the message m chosen by $\mathcal{A}$; the only restriction being that r and m must be of equal length. When the message space is $\mathcal{M} = \{0,1\}$, the above relation simplifies to

$$\Pr\left[\begin{array}{l}(\mathsf{mpk},\mathsf{msk}) \overset{R}{\leftarrow} \mathrm{SETUP}(1^\kappa),\\ (\mathsf{id}_0^*, \mathsf{id}_1^*, s) \leftarrow \mathcal{A}_1^{\mathrm{EXTRACT}_{\mathsf{msk}}(\cdot)}(\mathsf{mpk}),\\ b \overset{R}{\leftarrow} \{0,1\}, r \overset{R}{\leftarrow} \{0,1\}, C \leftarrow \mathrm{ENCRYPT}_{\mathsf{mpk}}(\mathsf{id}_b^*, r)\end{array} : \mathcal{A}_2^{\mathrm{EXTRACT}_{\mathsf{msk}}(\cdot)}(s, C) = b\right] - \frac{1}{2}.$$

Complexity assumptions

It is useful to introduce some notation. Let $N = pq$ be the product of two (odd) primes p and q. The Jacobi symbol modulo N of an integer a is denoted by $J_N(a)$. The set of integers whose Jacobi symbol is 1 is denoted by $\mathbb{J}_N$, $\mathbb{J}_N = \{a \in \mathbb{Z}_N^* | J_N(a) = 1\}$; the set of quadratic residues is denoted by $\mathbb{QR}_N$, $\mathbb{QR}_N = \{a \in \mathbb{Z}_N^* | J_p(a) = J_q(a) = 1\}$. Note that $\mathbb{QR}_N$ is a subset of $\mathbb{J}_N$.

Definition 1 (Quadratic Residuosity Assumption). Let RSAGen be a probabilistic algorithm which, given a security parameter κ, outputs primes p and q and their product $N = pq$. The *Quadratic Residuosity* (QR) assumption asserts that the success probability defined as the distance

$$\left|\Pr\left[\mathcal{D}(x, N) = 1 | x \overset{R}{\leftarrow} \mathbb{QR}_N\right] - \Pr\left[\mathcal{D}(x, N) = 1 | x \overset{R}{\leftarrow} \mathbb{J}_N \setminus \mathbb{QR}_N\right]\right|$$

is negligible for any probabilistic polynomial-time distinguisher $\mathcal{D}$; the probabilities are taken over the experiment of running $(N, p, q) \leftarrow \mathsf{RSAGen}(1^\kappa)$ and choosing at random $x \in \mathbb{QR}_N$ and $x \in \mathbb{J}_N \setminus \mathbb{QR}_N$.

Random oracle model The *random oracle model* is an idealized model introduced by Bellare and Rogaway [6] to analyze the security of certain cryptographic constructions using hash functions. Informally, the random oracle model assumes that the output of a hash function behaves as the output of a random generator.

The group $(\mathcal{F}_p)^2 \times (\mathcal{F}_q)^2$

Let p be an odd prime, let $\Delta \in \mathbb{F}_p{}^\times$, and let $\delta^2 = \Delta$. Define the multiplicative group

$$\mathsf{G}_p = \{x + \delta y | x, y \in \mathbb{F}_p \text{ and } x^2 - \Delta y^2 = 1\}.$$

Where it is defined, consider the map

$$\psi: \mathbb{F}_p \to \mathsf{G}_p, u \mapsto \frac{u + \delta}{u - \delta}.$$

To ease the notation, we augment $\mathbb{F}_p$ with a special symbol ∞ and define $\psi(\infty) = 1$. There are two cases to distinguish:

1. If $\delta \notin \mathbb{F}_p$ then $\mathsf{G}_p = \{\psi(u) | u \in \mathbb{F}_p \cup \{\infty\}\}$ —see [18];
2. If $\delta \in \mathbb{F}_p$ then $\mathsf{G}_p = \{\psi(u) | u \in \mathbb{F}_p \cup \{\infty\}, u \neq \pm\delta\}$.

(Observe that $0 \notin \mathsf{G}_p$ when $\delta \in \mathbb{F}_p$.)

We are interested in the second case. **From now on, we will suppose that $\delta \in \mathbb{F}_p{}^\times$ and thus that $\Delta \in \mathbb{QR}_p$.** For convenience, we write $\mathcal{F}_p = (\mathbb{F}_p \setminus \{\pm\delta\}) \cup$

$\{\infty\}$. We so have $\mathsf{G}_p = \psi(\mathcal{F}_p)$. Clearly, map $\psi: \mathcal{F}_p \to \mathsf{G}_p$ is injective and thus defines a bijection. Indeed, suppose $\psi(u_1) = \psi(u_2)$ for some $u_1, u_2 \in \mathcal{F}_p$. This implies $(u_1 + \delta)(u_2 - \delta) = (u_2 + \delta)(u_1 - \delta)$ and in turn $u_1 = u_2$. The inverse map is given by $\psi^{-1}: \mathsf{G}_p \to \mathcal{F}_p, v \mapsto \frac{\delta(v+1)}{v-1}$.

Furthermore, map ψ yields a homomorphism from $\mathcal{F}_p$ to G_p. We have $\psi(\infty) = 1$. Let $u_1, u_2 \in \mathcal{F}_p \setminus \{\infty\}$. We have:

$$\frac{1}{\psi(u_1)} = \frac{u_1 - \delta}{u_1 + \delta} = \frac{-u_1 + \delta}{-u_1 - \delta} = \psi(-u_1)$$

and, if $u_1 \neq -u_2$,

$$\psi(u_1) \cdot \psi(u_2) = \frac{(u_1 + \delta)(u_2 + \delta)}{(u_1 - \delta)(u_2 - \delta)} = \frac{u_1 u_2 + \Delta + \delta(u_1 + u_2)}{u_1 u_2 + \Delta - \delta(u_1 + u_2)} = \frac{\frac{u_1 u_2 + \Delta}{u_1 + u_2} + \delta}{\frac{u_1 u_2 + \Delta}{u_1 + u_2} - \delta}$$

$$= \psi(u_3)$$

where $u_3 = \frac{u_1 u_2 + \Delta}{u_1 + u_2}$.

Consequently, $\mathcal{F}_p$ endows a group structure. Its order is $\#\mathcal{F}_p = p - 1$. We write $\mathcal{F}_p$ multiplicatively and use $\circledast$ to denote its group law. We have:

- the neutral element is ∞:
$$u \circledast \infty = \infty \circledast u = u \quad \text{for all } u \in \mathcal{F}_p;$$

- the inverse of $u \in \mathcal{F}_p \setminus \{\infty\}$ is $-u$:
$$u \circledast (-u) = (-u) \circledast u = \infty;$$

- given $u_1, u_2 \in \mathcal{F}_p \setminus \{\infty\}$, their multiplication is given by:
$$u_1 \circledast u_2 = \begin{cases} \frac{u_1 u_2 + \Delta}{u_1 + u_2} & \text{if } u_1 \neq -u_2, \\ \infty & \text{otherwise.} \end{cases}$$

Remark that 0 is of order 2 as an element of $\mathcal{F}_p$, namely $0 \circledast 0 = \infty$. Remark also that for $u \in \mathcal{F}_p$, $u \neq 0, \infty$, we have $u \circledast 0 = \Delta/u$.

Let $(\mathcal{F}_p)^2 \subset \mathcal{F}_p$ represent the subgroup of squares in $\mathcal{F}_p$:

$$(\mathcal{F}_p)^2 = \{u \circledast u | u \in \mathcal{F}_p\}$$
$$= \{\frac{u^2 + \Delta}{2u} | u \in \mathbb{F}_p \setminus \{0, \pm\delta\}\} \cup \{\infty\}.$$

The group $(\mathcal{F}_p)^2$ has $(p-1)/2$ elements.

Let $u_1, u_2 \in \mathcal{F}_p$. Define $w_1 = u_1 \circledast u_1$ and $w_2 = u_2 \circledast u_2$. Then letting $u_3 := u_1 \circledast u_2$ we get:

$$w_3 := w_1 \circledast w_2$$
$$= (u_1 \circledast u_1) \circledast (u_2 \circledast u_2) = (u_1 \circledast u_2) \circledast (u_1 \circledast u_2) = u_3 \circledast u_3.$$

Further, for any $1 \leq i \leq 3$, if $w_i \neq \infty$ then

$$2w_i \pm 2\delta = \frac{(u_i \pm \delta)^2}{u_i}, \quad \text{and thus} \quad J_p(2w_i \pm 2\delta) = J_p(u_i). \tag{1}$$

The previous setting naturally extends through Chinese remaindering. Let $N = pq$ be an RSA (Rivest-Shamir-Adleman) modulus. Then $(\mathcal{F}_p)^2 \times (\mathcal{F}_q)^2$ is a group and has order $\phi(N)$.

SUMMARY OF THE INVENTION

The present invention recognizes the need to improve the existing systems and methods for implementing crytosystems and methods, especially anonymous cryptosystems and methods.

In accordance with an aspect of the present principles, a method for communicating a message m is presented, comprising:

accessing the message m;

accessing a user identity id and public system parameters $\mathsf{mpk} = \{N, u, \mathcal{H}, \{\mathcal{G}_{\mathsf{id}}\}, s\}$, where N is a composite integer, $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$, s is a bit-string, $\mathcal{H}$ is a cryptographic hash function mapping bit-strings to elements of $\mathbb{J}_N$ and $\{\mathcal{G}_{\mathsf{id}}\}$ is a family of functions mapping bit-strings to elements of a subset of $\mathbb{Z}/N\mathbb{Z}$, namely $\mathcal{G}_{\mathsf{id}}: \{0,1\}^* \to \mathbb{Z}/N\mathbb{Z}, s \mapsto d = \mathcal{G}_{\mathsf{id}}(s)$ such that $J_N(d^2 - 4\mathcal{H}(\mathsf{id})) = J_N(d^2 - 4u\mathcal{H}(\mathsf{id})) = -1$;

generating a ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ of the message m using the user identity id, where

$$\varepsilon = (-1)^m J_N(t), \quad c^{(0)} = t + \frac{R_{id}}{t} \bmod N, \quad c^{(1)} = \frac{c^{(0)} d + 4R_{id}}{c^{(0)} + d} \bmod N,$$

$$\bar{\varepsilon} = (-1)^m J_N(\bar{t}), \quad \bar{c}^{(0)} = \bar{t} + \frac{uR_{id}}{\bar{t}} \bmod N, \quad \bar{c}^{(1)} = \frac{\bar{c}^{(0)} d + 4uR_{id}}{\bar{c}^{(0)} + d} \bmod N,$$

where $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$, $t, \bar{t} \in (\mathbb{Z}/N\mathbb{Z})^\times$, $d = \mathcal{G}_{\mathsf{id}}(s)$, choosing random bits $\beta_1, \beta_2 \in \{0,1\}$ and setting $c = c^{(\beta_1)}$ and $\bar{c} = \bar{c}^{(\beta_2)}$; and

transmitting the ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ to a device via a communication channel.

In accordance with another aspect of the present principles, a method for processing a ciphertext is presented, comprising:

receiving the ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ for a user with an identity id via a communication channel; and

accessing the ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ of a plaintext message m, where

$$\varepsilon = (-1)^m J_N(t), \quad c^{(0)} = t + \frac{R_{id}}{t} \bmod N, \quad c^{(1)} = \frac{c^{(0)}d + 4R_{id}}{c^{(0)} + d} \bmod N,$$

$$\bar{\varepsilon} = (-1)^m J_N(\bar{t}), \quad \bar{c}^{(0)} = \bar{t} + \frac{uR_{id}}{\bar{t}} \bmod N, \quad \bar{c}^{(1)} = \frac{\bar{c}^{(0)}d + 4uR_{id}}{\bar{c}^{(0)} + d} \bmod N,$$

where N is a composite integer, $R_{\mathrm{id}} = \mathcal{H}(\mathrm{id})$, $t, \bar{t} \in (\mathbb{Z}/N\mathbb{Z})^{\times}$, $d = \mathcal{G}_{\mathrm{id}}(s)$, $c = c^{(\beta_1)}$ and $\bar{c} = \bar{c}^{(\beta_2)}$ for random bits $\beta_1, \beta_2 \in \{0,1\}$, wherein public system parameters $\mathsf{mpk} = \{N, u, \mathcal{H}, \{\mathcal{G}_{\mathrm{id}}\}, s\}$, where N is a composite integer, $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$, s is bit-string, $\mathcal{H}$ is a cryptographic hash function mapping bit-strings to elements of $\mathbb{J}_N$ and $\{\mathcal{G}_{\mathrm{id}}\}$ is a family of functions mapping bit-strings to elements of a subset of $\mathbb{Z}/N\mathbb{Z}$, namely $\mathcal{G}_{\mathrm{id}}: \{0,1\}^* \to \mathbb{Z}/N\mathbb{Z}, s \mapsto d = \mathcal{G}_{\mathrm{id}}(s)$ such that $J_N(d^2 - 4\mathcal{H}(\mathrm{id})) = J_N(d^2 - 4u\mathcal{H}(\mathrm{id})) = -1$;

generating the message m by accessing a private key $\mathsf{usk} = \{r_{\mathrm{id}}\}$ for a user with an identity id by following steps:

if ${r_{\mathrm{id}}}^2 \equiv R_{\mathrm{id}} \pmod N$, setting $\nu = \varepsilon$, $\gamma = c$ and $\Delta = R_{\mathrm{id}}$; otherwise setting $\nu = \bar{\varepsilon}$, $\gamma = \bar{c}$ and $\Delta = uR_{\mathrm{id}}$;

determining $\sigma = J_N(\gamma^2 - 4\Delta)$,

setting

$$\tau = \begin{cases} J_N(\gamma + 2r_{\mathrm{id}}) & \text{if } \sigma = 1 \\ J_N((\gamma + 2r_{\mathrm{id}})(d - 2r_{\mathrm{id}})(d - \gamma)) & \text{if } \sigma = -1 \end{cases}, \text{ where } d = \mathcal{G}_{\mathrm{id}}(s);$$

and

determining the message m as

$$m = \frac{1 - \nu \cdot \tau}{2}.$$

In accordance with another aspect of the present principles, a method is presented for generating keys for an identity-based cryptosystem wherein a private key is used to decrypt ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ of a plaintext message m for a user with an identity id where

$$\varepsilon = (-1)^m J_N(t), \quad c^{(0)} = t + \frac{R_{id}}{t} \bmod N, \quad c^{(1)} = \frac{c^{(0)}d + 4R_{id}}{c^{(0)} + d} \bmod N,$$

$$\bar{\varepsilon} = (-1)^m J_N(\bar{t}), \quad \bar{c}^{(0)} = \bar{t} + \frac{uR_{id}}{\bar{t}} \bmod N, \quad \bar{c}^{(1)} = \frac{\bar{c}^{(0)}d + 4uR_{id}}{\bar{c}^{(0)} + d} \bmod N,$$

where N is a composite integer, $R_{\mathrm{id}} = \mathcal{H}(\mathrm{id})$, $t, \bar{t} \in (\mathbb{Z}/N\mathbb{Z})^{\times}$, $d = \mathcal{G}_{\mathrm{id}}(s)$, $c = c^{(\beta_1)}$ and $\bar{c} = \bar{c}^{(\beta_2)}$ for random bits $\beta_1, \beta_2 \in \{0,1\}$, comprising steps of:

setting a master secret key msk; and

generating, using the master secret key msk, the private key $\mathsf{usk} = \{r_{\mathsf{id}}\}$ for a user with an identity id, using public system parameters $\mathsf{mpk} = \{N, u, \mathcal{H}, \{\mathcal{G}_{\mathsf{id}}\}, s\}$, where N is a composite integer, $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$, s is bit-string, $\mathcal{H}$ is a cryptographic hash function mapping bit-strings to elements of $\mathbb{J}_N$ and $\{\mathcal{G}_{\mathsf{id}}\}$ is a family of functions mapping bit-strings to elements of a subset of $\mathbb{Z}/N\mathbb{Z}$ namely $\mathcal{G}_{\mathsf{id}}: \{0,1\}^* \to \mathbb{Z}/N\mathbb{Z}, s \mapsto d = \mathcal{G}_{\mathsf{id}}(s)$ such that $J_N(d^2 - 4\mathcal{H}(\mathsf{id})) = J_N(d^2 - 4u\mathcal{H}(\mathsf{id})) = -1$.

In accordance with another aspect of the present principles, an apparatus is presented for communicating a message m, comprising:

a processor configured to access the message m, the processor further configured to access a user identity id and public system parameters $\mathsf{mpk} = \{N, u, \mathcal{H}, \{\mathcal{G}_{\mathsf{id}}\}, s\}$, where N is a composite integer, $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$, s is a bit-string, $\mathcal{H}$ is a cryptographic hash function mapping bit-strings to elements of $\mathbb{J}_N$ and $\{\mathcal{G}_{\mathsf{id}}\}$ is a family of functions mapping bit-strings to elements of a subset of $\mathbb{Z}/N\mathbb{Z}$, namely $\mathcal{G}_{\mathsf{id}}: \{0,1\}^* \to \mathbb{Z}/N\mathbb{Z}, s \mapsto d = \mathcal{G}_{\mathsf{id}}(s)$ such that $J_N(d^2 - 4\mathcal{H}(\mathsf{id})) = J_N(d^2 - 4u\mathcal{H}(\mathsf{id})) = -1$;

an encryptor configured to generate a ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ of the message m using the user identity id, where

$$\varepsilon = (-1)^m J_N(t), \quad c^{(0)} = t + \frac{R_{id}}{t} \bmod N, \quad c^{(1)} = \frac{c^{(0)} d + 4R_{id}}{c^{(0)} + d} \bmod N,$$

$$\bar{\varepsilon} = (-1)^m J_N(\bar{t}), \quad \bar{c}^{(0)} = \bar{t} + \frac{uR_{id}}{\bar{t}} \bmod N, \quad \bar{c}^{(1)} = \frac{\bar{c}^{(0)} d + 4uR_{id}}{\bar{c}^{(0)} + d} \bmod N,$$

where $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$, $t, \bar{t} \in (\mathbb{Z}/N\mathbb{Z})^\times$, $d = \mathcal{G}_{\mathsf{id}}(s)$, choosing random bits $\beta_1, \beta_2 \in \{0,1\}$ and setting $c = c^{(\beta_1)}$ and $\bar{c} = \bar{c}^{(\beta_2)}$; and

a communication interface, coupled to a communication channel, configured to transmit the ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ via the communication channel.

In accordance with another aspect of the present principles, an apparatus is presented for processing a ciphertext, comprising:

a communication interface, coupled to a communication channel, configured to receive the ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ of a plaintext message m, where

$$\varepsilon = (-1)^m J_N(t), \quad c^{(0)} = t + \frac{R_{id}}{t} \bmod N, \quad c^{(1)} = \frac{c^{(0)} d + 4R_{id}}{c^{(0)} + d} \bmod N,$$

$$\bar{\varepsilon} = (-1)^m J_N(\bar{t}), \quad \bar{c}^{(0)} = \bar{t} + \frac{uR_{id}}{\bar{t}} \bmod N, \quad \bar{c}^{(1)} = \frac{\bar{c}^{(0)}d + 4uR_{id}}{\bar{c}^{(0)} + d} \bmod N,$$

where N is a composite integer, $R_{\mathrm{id}} = \mathcal{H}(\mathrm{id})$, $t, \bar{t} \in (\mathbb{Z}/N\mathbb{Z})^{\times}$, $d = \mathcal{G}_{\mathrm{id}}(s)$, $c = c^{(\beta_1)}$ and $\bar{c} = \bar{c}^{(\beta_2)}$ for random bits $\beta_1, \beta_2 \in \{0,1\}$, wherein public system parameters $\mathrm{mpk} = \{N, u, \mathcal{H}, \{\mathcal{G}_{\mathrm{id}}\}, s\}$, where N is a composite integer, $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$, s is bit-string, $\mathcal{H}$ is a cryptographic hash function mapping bit-strings to elements of $\mathbb{J}_N$ and $\{\mathcal{G}_{\mathrm{id}}\}$ is a family of functions mapping bit-strings to elements of a subset of $\mathbb{Z}/N\mathbb{Z}$, namely $\mathcal{G}_{\mathrm{id}}: \{0,1\}^* \to \mathbb{Z}/N\mathbb{Z}, s \mapsto d = \mathcal{G}_{\mathrm{id}}(s)$ such that $J_N(d^2 - 4\mathcal{H}(\mathrm{id})) = J_N(d^2 - 4u\mathcal{H}(\mathrm{id})) = -1$; and

the processor generates the plaintext message m by accessing a private key $\mathrm{usk} = \{r_{\mathrm{id}}\}$ for a user with an identity id by following steps:

if ${r_{\mathrm{id}}}^2 \equiv R_{\mathrm{id}} \pmod N$, setting $\nu = \varepsilon$, $\gamma = c$ and $\Delta = R_{\mathrm{id}}$; otherwise setting $\nu = \bar{\varepsilon}$, $\gamma = \bar{c}$ and $\Delta = uR_{\mathrm{id}}$;

determining $\sigma = J_N(\gamma^2 - 4\Delta)$,

setting

$$\tau = \begin{cases} J_N(\gamma + 2r_{\mathrm{id}}) & \text{if } \sigma = 1 \\ J_N((\gamma + 2r_{\mathrm{id}})(d - 2r_{\mathrm{id}})(d - \gamma)) & \text{if } \sigma = -1 \end{cases}, \text{ where } d = \mathcal{G}_{\mathrm{id}}(s);$$

and

determining the plaintext m as

$$m = \frac{1 - \nu \cdot \tau}{2}.$$

In accordance with another aspect of the present principles, an apparatus is presented for generating keys for an identity-based cryptosystem wherein a private key is used to decrypt ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ of a plaintext message m for a user with an identity id where

$$\varepsilon = (-1)^m J_N(t), \quad c^{(0)} = t + \frac{R_{id}}{t} \bmod N, \quad c^{(1)} = \frac{c^{(0)}d + 4R_{id}}{c^{(0)} + d} \bmod N,$$

$$\bar{\varepsilon} = (-1)^m J_N(\bar{t}), \quad \bar{c}^{(0)} = \bar{t} + \frac{uR_{id}}{\bar{t}} \bmod N, \quad \bar{c}^{(1)} = \frac{\bar{c}^{(0)}d + 4uR_{id}}{\bar{c}^{(0)} + d} \bmod N,$$

where N is a composite integer, $R_{\mathrm{id}} = \mathcal{H}(\mathrm{id})$, $t, \bar{t} \in (\mathbb{Z}/N\mathbb{Z})^{\times}$, $d = \mathcal{G}_{\mathrm{id}}(s)$, $c = c^{(\beta_1)}$ and $\bar{c} = \bar{c}^{(\beta_2)}$ for random bits $\beta_1, \beta_2 \in \{0,1\}$, comprising:

a processor configured to set a master secret key msk, and generate, using the master secret key msk, the private key $\mathrm{usk} = \{r_{\mathrm{id}}\}$ for a user with an identity id, using

public system parameters $\mathsf{mpk} = \{N, u, \mathcal{H}, \{\mathcal{G}_{\mathsf{id}}\}, s\}$, where N is a composite integer, $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$, s is bit-string, $\mathcal{H}$ is a cryptographic hash function mapping bit-strings to elements of $\mathbb{J}_N$ and $\{\mathcal{G}_{\mathsf{id}}\}$ is a family of functions mapping bit-strings to elements of a subset of $\mathbb{Z}/N\mathbb{Z}$ namely $\mathcal{G}_{\mathsf{id}}: \{0,1\}^* \to \mathbb{Z}/N\mathbb{Z}, s \mapsto d = \mathcal{G}_{\mathsf{id}}(s)$ such that $J_N(d^2 - 4\mathcal{H}(\mathsf{id})) = J_N(d^2 - 4u\mathcal{H}(\mathsf{id})) = -1$; and

a communication interface, coupled to a communication channel, configured to transmit the private key $\mathsf{usk} = \{r_{\mathsf{id}}\}$ via the communication channel.

In accordance with another aspect of the present principles, a computer program product stored in non-transitory computer-readable storage media is presented, comprising computer-executable instructions for

accessing the message m;

accessing a user identity id and public system parameters $\mathsf{mpk} = \{N, u, \mathcal{H}, \{\mathcal{G}_{\mathsf{id}}\}, s\}$, where N is a composite integer, $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$, s is a bit-string, $\mathcal{H}$ is a cryptographic hash function mapping bit-strings to elements of $\mathbb{J}_N$ and $\{\mathcal{G}_{\mathsf{id}}\}$ is a family of functions mapping bit-strings to elements of a subset of $\mathbb{Z}/N\mathbb{Z}$, namely $\mathcal{G}_{\mathsf{id}}: \{0,1\}^* \to \mathbb{Z}/N\mathbb{Z}, s \mapsto d = \mathcal{G}_{\mathsf{id}}(s)$ such that $J_N(d^2 - 4\mathcal{H}(\mathsf{id})) = J_N(d^2 - 4u\mathcal{H}(\mathsf{id})) = -1$;

generating a ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ of the message m using the user identity id, where

$$\varepsilon = (-1)^m J_N(t), \quad c^{(0)} = t + \frac{R_{id}}{t} \bmod N, \quad c^{(1)} = \frac{c^{(0)} d + 4R_{id}}{c^{(0)} + d} \bmod N,$$

$$\bar{\varepsilon} = (-1)^m J_N(\bar{t}), \quad \bar{c}^{(0)} = \bar{t} + \frac{uR_{id}}{\bar{t}} \bmod N, \quad \bar{c}^{(1)} = \frac{\bar{c}^{(0)} d + 4uR_{id}}{\bar{c}^{(0)} + d} \bmod N,$$

where $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$, $t, \bar{t} \in (\mathbb{Z}/N\mathbb{Z})^\times$, $d = \mathcal{G}_{\mathsf{id}}(s)$, choosing random bits $\beta_1, \beta_2 \in \{0,1\}$ and setting $c = c^{(\beta_1)}$ and $\bar{c} = \bar{c}^{(\beta_2)}$; and

transmitting the ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ to a device via a communication channel.

In accordance with another aspect of the present principles, a computer program product stored in non-transitory computer-readable storage media is presented, comprising computer-executable instructions for

receiving the ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ for a user with an identity id via a communication channel; and

accessing the ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ of a plaintext message m, where

$$\varepsilon = (-1)^m J_N(t), \quad c^{(0)} = t + \frac{R_{id}}{t} \bmod N, \quad c^{(1)} = \frac{c^{(0)}d + 4R_{id}}{c^{(0)} + d} \bmod N,$$

$$\bar{\varepsilon} = (-1)^m J_N(\bar{t}), \quad \bar{c}^{(0)} = \bar{t} + \frac{uR_{id}}{\bar{t}} \bmod N, \quad \bar{c}^{(1)} = \frac{\bar{c}^{(0)}d + 4uR_{id}}{\bar{c}^{(0)} + d} \bmod N,$$

where N is a composite integer, $R_{\mathrm{id}} = \mathcal{H}(\mathrm{id})$, $t, \bar{t} \in (\mathbb{Z}/N\mathbb{Z})^{\times}$, $d = \mathcal{G}_{\mathrm{id}}(s)$, $c = c^{(\beta_1)}$ and $\bar{c} = \bar{c}^{(\beta_2)}$ for random bits $\beta_1, \beta_2 \in \{0,1\}$, wherein public system parameters $\mathsf{mpk} = \{N, u, \mathcal{H}, \{\mathcal{G}_{\mathrm{id}}\}, s\}$, where N is a composite integer, $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$, s is bit-string, $\mathcal{H}$ is a cryptographic hash function mapping bit-strings to elements of $\mathbb{J}_N$ and $\{\mathcal{G}_{\mathrm{id}}\}$ is a family of functions mapping bit-strings to elements of a subset of $\mathbb{Z}/N\mathbb{Z}$, namely $\mathcal{G}_{\mathrm{id}}: \{0,1\}^* \to \mathbb{Z}/N\mathbb{Z}, s \mapsto d = \mathcal{G}_{\mathrm{id}}(s)$ such that $J_N(d^2 - 4\mathcal{H}(\mathrm{id})) = J_N(d^2 - 4u\mathcal{H}(\mathrm{id})) = -1$;

generating the message m by accessing a private key $\mathsf{usk} = \{r_{\mathrm{id}}\}$ for a user with an identity id by following steps:

if $r_{\mathrm{id}}{}^2 \equiv R_{\mathrm{id}} \pmod N$, setting $\nu = \varepsilon$, $\gamma = c$ and $\Delta = R_{\mathrm{id}}$; otherwise setting $\nu = \bar{\varepsilon}$, $\gamma = \bar{c}$ and $\Delta = uR_{\mathrm{id}}$;

determining $\sigma = J_N(\gamma^2 - 4\Delta)$,

setting

$$\tau = \begin{cases} J_N(\gamma + 2r_{\mathrm{id}}) & \text{if } \sigma = 1 \\ J_N((\gamma + 2r_{\mathrm{id}})(d - 2r_{\mathrm{id}})(d - \gamma)) & \text{if } \sigma = -1 \end{cases}, \text{ where } d = \mathcal{G}_{\mathrm{id}}(s);$$

and

determining the message m as

$$m = \frac{1 - \nu \cdot \tau}{2}.$$

In accordance with another aspect of the present principles, a computer program product stored in non-transitory computer-readable storage media is presented, for generating keys for an identity-based cryptosystem wherein a private key is used to decrypt ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ of a plaintext message m for a user with an identity id where

$$\varepsilon = (-1)^m J_N(t), \quad c^{(0)} = t + \frac{R_{id}}{t} \bmod N, \quad c^{(1)} = \frac{c^{(0)}d + 4R_{id}}{c^{(0)} + d} \bmod N,$$

$$\bar{\varepsilon} = (-1)^m J_N(\bar{t}), \quad \bar{c}^{(0)} = \bar{t} + \frac{uR_{id}}{\bar{t}} \bmod N, \quad \bar{c}^{(1)} = \frac{\bar{c}^{(0)}d + 4uR_{id}}{\bar{c}^{(0)} + d} \bmod N,$$

where N is a composite integer, $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$, $t, \bar{t} \in (\mathbb{Z}/N\mathbb{Z})^{\times}$, $d = \mathcal{G}_{\mathsf{id}}(s)$, $c = c^{(\beta_1)}$ and $\bar{c} = \bar{c}^{(\beta_2)}$ for random bits $\beta_1, \beta_2 \in \{0,1\}$, comprising computer-executable instructions for:

setting a master secret key msk; and

generating, using the master secret key msk, the private key $\mathsf{usk} = \{r_{\mathsf{id}}\}$ for a user with an identity id, using public system parameters $\mathsf{mpk} = \{N, u, \mathcal{H}, \{\mathcal{G}_{\mathsf{id}}\}, s\}$, where N is a composite integer, $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$, s is bit-string, $\mathcal{H}$ is a cryptographic hash function mapping bit-strings to elements of $\mathbb{J}_N$ and $\{\mathcal{G}_{\mathsf{id}}\}$ is a family of functions mapping bit-strings to elements of a subset of $\mathbb{Z}/N\mathbb{Z}$ namely $\mathcal{G}_{\mathsf{id}}: \{0,1\}^* \to \mathbb{Z}/N\mathbb{Z}, s \mapsto d = \mathcal{G}_{\mathsf{id}}(s)$ such that $J_N(d^2 - 4\mathcal{H}(\mathsf{id})) = J_N(d^2 - 4u\mathcal{H}(\mathsf{id})) = -1$.

DETAILED DESCRIPTION OF THE DRAWINGS

The above-mentioned and other features and advantages of this invention, and the manner of attaining them, will become more apparent and the invention will be better understood by reference to the following description of embodiments of the invention taken in conjunction with the accompanying drawings, wherein:

Figures 1 to 3 show exemplary apparatus according to the present principles; and

Figures 4 to 6 show exemplary processes according to the present principles.

The examples set out herein illustrate exemplary embodiments of the invention. Such examples are not to be construed as limiting the scope of the invention in any manner.

DETAILED DESCRIPTION

Technical problems to solve

In numerous scenarios, the recipient in a transmission needs to be kept *anonymous*. This allows users to maintain some privacy. Protecting communication content may be not enough, as already observed in a couple of papers (e.g., [4,5,17]). For example, by analyzing the traffic between an antenna and a mobile device, one can recover some information about [at least] user's position and some details about the use of her mobile device. This information leaks easily during all day: it is a

common habit, indeed, to use a mobile phone every day and to keep it (almost) always switched on.

There are a couple of anonymous IBE schemes based on quadratic residuosity in the literature [9,11,3,10]. Among them the most efficient ones are those of Ateniese and Gasti [3] and the recent one by Clear *et al.* [10]. Table 1 in [10] gives a comparison of the schemes. The scheme by Clear *et al.* features the best encryption and decryption times (i.e., 79 ms and 27 ms for a 128-message with a key-size of 1024 bits in their setting). The security relies on the quadratic residuosity assumption in the random oracle model.

However, the cryptosystem by Clear *et al.* produces somewhat long ciphertexts. The goal of the present principles is to provide anonymous *identity-based* cryptosystems with much shorter ciphertexts. As will be seen, the resulting ciphertexts are twice shorter, improving all previous anonymous IBEs based on the quadratic residuosity. Extra useful properties offered by the proposed cryptosystems are listed later.

Main ideas

As a reminder, using the notation described before, $\mathcal{F}_p = \{u | u \in \mathbb{F}_p \setminus \{\pm\delta\}\} \cup \{\infty\}$ and $\mathsf{G}_p = \{v = x + \delta y | x, y \in \mathbb{F}_p, x^2 - \Delta y^2 = 1\}$ are groups under the respective laws $\circledast$ and $\cdot$ (multiplication in $\mathbb{F}_p$); the respective neutral element are ∞ and 1. Finally, $\psi: \mathcal{F}_p \xrightarrow{\sim} \mathsf{G}_p, u \mapsto v = \psi(u)$ is a group isomorphism. Specifically, we have $\psi(\infty) = 1$ and $\psi(u) = \frac{u+\delta}{u-\delta}$ if $u \neq \infty$. We also define the subgroup of squares in $\mathcal{F}_p$, namely $(\mathcal{F}_p)^2 = \{u \circledast u | u \in \mathcal{F}_p\}$. Alternatively, as shown in [14], this group can be defined by

$$(\mathcal{F}_p)^2 = \{u \in \mathbb{F}_p \setminus \{\pm\delta\} | J_p(u^2 - \Delta) = 1\} \cup \{\infty\}.$$

Conversely, the group of non-squares in $\mathcal{F}_p$ can be defined as

$$(\mathcal{F}_p)^{\neq 2} = \{u \in \mathbb{F}_p \setminus \{\pm\delta\} | J_p(u^2 - \Delta) = -1\} \cup \{\infty\}.$$

The next proposition [14] is central to our ideas.

Proposition 1. *Let $N = pq$ be an RSA modulus and let $w \in (\mathcal{F}_p \setminus \{\infty\}) \times (\mathcal{F}_q \setminus \{\infty\})$. If*

$$J_N(w^2 - \Delta) = -1$$

then $w \notin (\mathcal{F}_p)^2 \times (\mathcal{F}_q)^2$.

Let $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$. It implies that elements of the form $c = \frac{t^2+R_{id}}{2t} \bmod N$ (respectively, $\bar{c} = \frac{\bar{t}^2+uR_{id}}{2\bar{t}} \bmod N$) (or multiples thereof) —where $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id}) \in \mathbb{QR}_N$ (respectively, $uR_{\mathsf{id}} = u\mathcal{H}(\mathsf{id}) \in \mathbb{QR}_N$) is derived from some user's identity id— cannot be used as part of a ciphertext for user with identity id' because if $J_N(c^2 - R_{\mathsf{id}'}) = -1$ (respectively, if $J_N(\bar{c}^2 - uR_{\mathsf{id}'}) = -1$) —where $R_{\mathsf{id}'} = \mathcal{H}(\mathsf{id}') \in \mathbb{QR}_N$ (respectively, $uR_{\mathsf{id}'} = u\mathcal{H}(\mathsf{id}') \in \mathbb{QR}_N$) for some other identity id'— then one can conclude that the identity of the recipient of the ciphertext is <u>not</u> id'. This clearly violates the anonymity requirement.

In order to address this issue, we propose to $\circledast$-multiply with probability $1/2$ the value of c (resp. $\bar{c}$) by an element d satisfying $J_N(d^2 - \Delta) = -1$ (resp. $J_N(d^2 - u\Delta) = -1$). The decryption algorithm, assuming it is the legitimate recipient of the ciphertext, can then $\circledast$-divide by d the ciphertext in the case it was $\circledast$-multiplied by d; letting e (resp. $\bar{e}$) the received part of the ciphertext, it is easy for the decryption algorithm to know if $e = c$ or $e = c \circledast d$ (resp. $\bar{e} = \bar{c}$ or $\bar{e} = \bar{c} \circledast d$) by checking if $J_N(e^2 - \Delta) = 1$ or -1 (resp. $J_N(\bar{e}^2 - u\Delta) = 1$ or -1), respectively.

In a typical implementation, parameter d is derived from the system parameters and/or the identity of the recipient.

Remark 2. Instead of $\circledast$-dividing by d, one could also $\circledast$-multiply by d. The resulting element would then be in $(\mathcal{F}_p)^2 \times (\mathcal{F}_q)^2$, as desired. A close inspection shows that the corresponding decryption process is likely less efficient.

Exemplary embodiments

First exemplary embodiment

The exemplary embodiments are now described in the framework of the flow diagram of Figure 3. Our first cryptosystem is defined as follows. For slightly better performance, all elements of $\mathcal{F}_p \times \mathcal{F}_q$ are scaled by a factor of 2.

SETUP(1^κ) Given a security parameter κ, SETUP generates an RSA modulus $N = pq$ where p and q are prime. It selects a bit-string s. It also generates a random element $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$. The public system parameters are $\mathsf{mpk} = \{N, u, \mathcal{H}, \{\mathcal{G}_{\mathsf{id}}\}, s\}$ where $\mathcal{H}$ is a cryptographic hash function mapping bit-strings to elements of $\mathbb{J}_N$ (i.e.,

$\mathcal{H}: \{0,1\}^* \to \mathbb{J}_N$) and $\{\mathcal{G}_{\mathsf{id}}\}$ is a family of functions mapping bit-strings to elements of a subset of $\mathbb{Z}/N\mathbb{Z}$, namely

$$\mathcal{G}_{\mathsf{id}}: \{0,1\}^* \to \mathbb{Z}/N\mathbb{Z}, s \mapsto d = \mathcal{G}_{\mathsf{id}}(s)$$
$$\text{such that } J_N(d^2 - 4\mathcal{H}(\mathsf{id})) = J_N(d^2 - 4u\mathcal{H}(\mathsf{id})) = -1.$$

The master secret key is $\mathsf{msk} = \{p, q\}$.

$\mathrm{EXTRACT}_{\mathsf{msk}}(\mathsf{id})$ Given identity id, key derivation algorithm EXTRACT first sets $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$. If $R_{\mathsf{id}} \in \mathbb{QR}_N$ it computes $r_{\mathsf{id}} = {R_{\mathsf{id}}}^{1/2} \mod N$; otherwise it computes $r_{\mathsf{id}} = (uR_{\mathsf{id}})^{1/2} \mod N$. EXTRACT returns user's private key $\mathsf{usk} = \{r_{\mathsf{id}}\}$.

$\mathrm{ENCRYPT}_{\mathsf{mpk}}(\mathsf{id}, m)$ To encrypt a message $m \in \{0,1\}$ for a user with identity id, ENCRYPT first defines $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$. It chooses at random $t, \bar{t} \in (\mathbb{Z}/N\mathbb{Z})^\times$ satisfying $\gcd(t^2 - R_{\mathsf{id}}, N) = \gcd(\bar{t}^2 - uR_{\mathsf{id}}, N) = 1$ and lets

$$\varepsilon = (-1)^m J_N(t), \quad c^{(0)} = t + \frac{R_{id}}{t} \mod N, \quad c^{(1)} = \frac{c^{(0)}d + 4R_{id}}{c^{(0)} + d} \mod N,$$

$$\bar{\varepsilon} = (-1)^m J_N(\bar{t}), \quad \bar{c}^{(0)} = \bar{t} + \frac{uR_{id}}{\bar{t}} \mod N, \quad \bar{c}^{(1)} = \frac{\bar{c}^{(0)}d + 4uR_{id}}{\bar{c}^{(0)} + d} \mod N,$$

where $d = \mathcal{G}_{\mathsf{id}}(s)$. It chooses random bits $\beta_1, \beta_2 \in \{0,1\}$ and sets $c = c^{(\beta_1)}$ and $\bar{c} = \bar{c}^{(\beta_2)}$. The returned ciphertext is $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$.

$\mathrm{DECRYPT}_{\mathsf{usk}}(C)$ Let $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$. From $\mathsf{usk} = \{r_{\mathsf{id}}\}$ and $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$, if ${r_{\mathsf{id}}}^2 \equiv R_{\mathsf{id}} \pmod N$, DECRYPT sets $\nu = \varepsilon$, $\gamma = c$ and $\Delta = R_{\mathsf{id}}$; otherwise it sets $\nu = \bar{\varepsilon}$, $\gamma = \bar{c}$ and $\Delta = uR_{\mathsf{id}}$. Next, it computes $\sigma = J_N(\gamma^2 - 4\Delta)$. If $\sigma \notin \{\pm 1\}$ it returns $\perp$; otherwise, from $d = \mathcal{G}_{\mathsf{id}}(s)$, it sets

$$\tau = \begin{cases} J_N(\gamma + 2r_{\mathsf{id}}) & \text{if } \sigma = 1 \\ J_N((\gamma + 2r_{\mathsf{id}})(d - 2r_{\mathsf{id}})(d - \gamma)) & \text{if } \sigma = -1 \end{cases}$$

and returns plaintext m as

$$m = \frac{1 - \nu \cdot \tau}{2}.$$

Remark 3. As an alternative, we see from Eq. (1) that the decryption algorithm can evaluate τ as $\tau = J_N(\gamma - 2r_{\mathsf{id}})$ when $\sigma = 1$ and as $\tau = J_N((\gamma - 2r_{\mathsf{id}})(d + 2r_{\mathsf{id}})(d - \gamma)) = -J_N((\gamma - 2r_{\mathsf{id}})(d_{\mathsf{id}} - 2r_{\mathsf{id}})(d_{\mathsf{id}} - \gamma)) = -J_N((\gamma + 2r_{\mathsf{id}})(d + 2r_{\mathsf{id}})(d - \gamma))$ when $\sigma = -1$. Also the value of $J_N(d + 2r_{\mathsf{id}})$ or of $J_N(d - 2r_{\mathsf{id}})$ can be precomputed.

Remark 4. Input s can be the empty string in the encryption/decryption algorithm.

Remark 5. In a practical implementation, for better efficiency, the encryption algorithm can randomly draw $t, \bar{t} \in \mathbb{Z}/N\mathbb{Z}$.

Correctness

Correctness is straightforward and follows from [14].

Security analysis

The two next propositions assess the security of the scheme under the quadratic residuosity assumption, in the random oracle model.

Proposition 2. *The scheme is* IND-ID-CPA *under the quadratic residuosity assumption in the random oracle model.*

Proof. Analogously to the proof of [14, Proposition 2] this would imply an IND-ID-CPA adversary against the scheme of [15, first exemplary embodiment].

Proposition 3. The scheme is ANO-ID-CPA under the quadratic residuosity assumption in the random oracle model.

Proof. The proof can be derived from [14, Proposition 3].

The two propositions show that the scheme meets the ANO-IND-ID-CPA under the quadratic residuosity assumption in the random oracle model.

Second exemplary embodiment

Let ℓ represent the bit-length of RSA modulus N. Similarly to [14], the previous scheme can be adapted so that the ciphertext needs at most 2ℓ bits for its representation —the previous implementation needs $2\ell + 2$ bits. Here is an implementation of a so-obtained cryptosystem.

SETUP(1^κ) Given a security parameter κ, SETUP generates an RSA modulus $N = pq$ where p and q are prime. It selects a bit-string s. It also generates a random element $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$. The public system parameters are $\mathsf{mpk} = \{N, u, \mathcal{H}, \{\mathcal{G}_{\mathsf{id}}\}, s\}$ where $\mathcal{H}$ is a cryptographic hash function mapping bit-strings to elements of $\mathbb{J}_N$ (i.e., $\mathcal{H}: \{0,1\}^* \to \mathbb{J}_N$) and $\{\mathcal{G}_{\mathsf{id}}\}$ is a family of functions mapping bit-strings to elements of a subset of $\mathbb{Z}/N\mathbb{Z}$, namely

$$\mathcal{G}_{\mathsf{id}}: \{0,1\}^* \to \mathbb{Z}/N\mathbb{Z}, \sigma \mapsto d = \mathcal{G}_{\mathsf{id}}(s) \text{ such that } J_N(d^2 - 4\mathcal{H}(\mathsf{id})) = J_N(d^2 - 4u\mathcal{H}(\mathsf{id})) = -1.$$

The master secret key is $\mathsf{msk} = \{p, q\}$.

$\mathrm{EXTRACT}_{\mathsf{msk}}(\mathsf{id})$ Given identity id, key derivation algorithm EXTRACT first sets $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$. If $R_{\mathsf{id}} \in \mathbb{QR}_N$ it computes $r_{\mathsf{id}} = {R_{\mathsf{id}}}^{1/2} \mod N$; otherwise it computes $r_{\mathsf{id}} = (uR_{\mathsf{id}})^{1/2} \mod N$. EXTRACT returns user's private key $\mathsf{usk} = \{r_{\mathsf{id}}\}$.

$\mathrm{ENCRYPT}_{\mathsf{mpk}}(\mathsf{id}, m)$ To encrypt a message $m \in \{0,1\}$ for a user with identity id, ENCRYPT first defines $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$. It chooses at random $t, \bar{t} \in \mathbb{Z}/N\mathbb{Z}$ and lets

$$\varepsilon' = (-1)^m J_N(t), \quad c'^{(0)} = t + \frac{R_{id}}{t} \mod N, \quad c'^{(1)} = \frac{c'^{(0)} d + 4R_{id}}{c'^{(0)} + d} \mod N,$$

$$\bar{\varepsilon}' = (-1)^m J_N(\bar{t}), \quad \bar{c}'^{(0)} = \bar{t} + \frac{uR_{id}}{\bar{t}} \mod N, \quad \bar{c}'^{(1)} = \frac{\bar{c}'^{(0)} d + 4uR_{id}}{\bar{c}'^{(0)} + d} \mod N,$$

where $d = \mathcal{G}_{\mathsf{id}}(s)$. It chooses random bits $\beta_1, \beta_2 \in \{0,1\}$ and sets $c' = c'^{(\beta_1)}$ and $\bar{c}' = \bar{c}'^{(\beta_2)}$. Define $c = \min(c', N - c')$ and $\bar{c} = \min(\bar{c}', N - \bar{c}')$. If $c = c'$ then define $\varepsilon = \varepsilon'$; otherwise define

$$\varepsilon = \begin{cases} \varepsilon' \cdot J_N(-1) & \text{if } \beta_1 = 0 \\ \varepsilon' \cdot J_N(-(td + 2R_{\mathsf{id}})(td + 2t^2)) & \text{if } \beta_1 = 1 \end{cases}.$$

Likewise, if $\bar{c} = \bar{c}'$ then define $\bar{\varepsilon} = \bar{\varepsilon}'$; otherwise define

$$\bar{\varepsilon} = \begin{cases} \bar{\varepsilon}' \cdot J_N(-1) & \text{if } \beta_2 = 0 \\ \bar{\varepsilon}' \cdot J_N(-(\bar{t}d + 2uR_{\mathsf{id}})(\bar{t}d + 2\bar{t}^2)) & \text{if } \beta_2 = 1 \end{cases}.$$

The returned ciphertext is $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$.

$\mathrm{DECRYPT}_{\mathsf{usk}}(C)$ Let $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$. From $\mathsf{usk} = \{r_{\mathsf{id}}\}$ and $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$, if ${r_{\mathsf{id}}}^2 \equiv R_{\mathsf{id}} \pmod N$, DECRYPT sets $\nu = \varepsilon$, $\gamma = c$ and $\Delta = R_{\mathsf{id}}$; otherwise it sets $\nu = \bar{\varepsilon}$, $\gamma = \bar{c}$ and $\Delta = uR_{\mathsf{id}}$. Next, it computes $\sigma = J_N(\gamma^2 - 4\Delta)$. If $\sigma \notin \{\pm 1\}$ it returns $\perp$; otherwise, from $d = \mathcal{G}_{\mathsf{id}}(s)$, it sets

$$\tau = \begin{cases} J_N(\gamma + 2r_{\mathsf{id}}) & \text{if } \sigma = 1 \\ J_N((\gamma + 2r_{\mathsf{id}})(d - 2r_{\mathsf{id}})(d - \gamma)) & \text{if } \sigma = -1 \end{cases}$$

and returns plaintext m as

$$m = \frac{1 - \nu \cdot \tau}{2}.$$

Remark 6. For slightly better efficiency, as a variant, the encryption algorithm can directly evaluate ε as $\varepsilon = (-1)^m J_N(-(td + 2R_{\mathsf{id}})(d + 2t))$ when $c \neq c'$ and

$\beta_1 = 1$. Likewise, it can evaluate $\bar{\varepsilon}$ as $\bar{\varepsilon} = (-1)^m J_N(-(\bar{t}d + 2uR_{\mathsf{id}})(d + 2\bar{t}))$ when $\bar{c} \neq \bar{c}'$ and $\beta_2 = 1$.

Remark 7. As an alternative, the decryption algorithm can evaluate τ as $\tau = J_N(\gamma - 2r_{\mathsf{id}})$ when $\sigma = 1$ and as $\tau = J_N((\gamma - 2r_{\mathsf{id}})(d + 2r_{\mathsf{id}})(d - \gamma)) = -J_N((\gamma - 2r_{\mathsf{id}})(d_{\mathsf{id}} - 2r_{\mathsf{id}})(d_{\mathsf{id}} - \gamma)) = -J_N((\gamma + 2r_{\mathsf{id}})(d + 2r_{\mathsf{id}})(d - \gamma))$ when $\sigma = -1$. Also the value of $J_N(d + 2r_{\mathsf{id}})$ or of $J_N(d - 2r_{\mathsf{id}})$ can be precomputed.

Third exemplary embodiment

Instead of selecting parameter d as the output of a family of functions, it can be chosen as, for example, the smallest nonnegative integer d such that $J_N(d^2 - 4\mathcal{H}(\mathsf{id})) = J_N(d^2 - 4u\mathcal{H}(\mathsf{id})) = -1$. We give below an illustration with the first exemplary embodiment. A similar embodiment can be obtained for the second exemplary embodiment.

The advantage is that the system parameters are smaller since $\{\mathcal{G}_{\mathsf{id}}\}$ is not explicitly included in mpk.

SETUP(1^κ) Given a security parameter κ, SETUP generates an RSA modulus $N = pq$ where p and q are prime. It also generates a random element $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$. The public system parameters are $\mathsf{mpk} = \{N, u, \mathcal{H}\}$ where $\mathcal{H}$ is a cryptographic hash function mapping bit-strings to elements of $\mathbb{J}_N$. The master secret key is $\mathsf{msk} = \{p, q\}$.

EXTRACT$_{\mathsf{msk}}(\mathsf{id})$ Given identity id, key derivation algorithm EXTRACT first sets $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$. If $R_{\mathsf{id}} \in \mathbb{QR}_N$ it computes $r_{\mathsf{id}} = {R_{\mathsf{id}}}^{1/2} \mod N$; otherwise it computes $r_{\mathsf{id}} = (uR_{\mathsf{id}})^{1/2} \mod N$. EXTRACT returns user's private key $\mathsf{usk} = \{r_{\mathsf{id}}\}$.

ENCRYPT$_{\mathsf{mpk}}(\mathsf{id}, m)$ To encrypt a message $m \in \{0,1\}$ for a user with identity id, ENCRYPT first defines $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$. Next, it tries $d = 0,1,2,\ldots$ until $J_N(d^2 - 4R_{\mathsf{id}}) = J_N(d^2 - 4uR_{\mathsf{id}}) = -1$. It chooses at random $t, \bar{t} \in \mathbb{Z}/N\mathbb{Z}$ and lets

$$\varepsilon = (-1)^m J_N(t), \quad c^{(0)} = t + \frac{R_{id}}{t} \mod N, \quad c^{(1)} = \frac{c^{(0)}d + 4R_{id}}{c^{(0)} + d} \mod N,$$

$$\bar{\varepsilon} = (-1)^m J_N(\bar{t}), \quad \bar{c}^{(0)} = \bar{t} + \frac{uR_{id}}{\bar{t}} \mod N, \quad \bar{c}^{(1)} = \frac{\bar{c}^{(0)}d + 4uR_{id}}{\bar{c}^{(0)} + d} \mod N.$$

It chooses random bits $\beta_1, \beta_2 \in \{0,1\}$ and sets $c = c^{(\beta_1)}$ and $\bar{c} = \bar{c}^{(\beta_2)}$. The returned ciphertext is $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$.

DECRYPT$_{\mathsf{usk}}(C)$ Let $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$. From $\mathsf{usk} = \{r_{\mathsf{id}}\}$ and $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$, if ${r_{\mathsf{id}}}^2 \equiv R_{\mathsf{id}} \pmod N$, DECRYPT sets $\nu = \varepsilon$, $\gamma = c$ and $\Delta = R_{\mathsf{id}}$; otherwise it sets

$v = \bar{\varepsilon}$, $\gamma = \bar{c}$ and $\Delta = uR_{\mathsf{id}}$. Next, it computes $\sigma = J_N(\gamma^2 - 4\Delta)$. If $\sigma \notin \{\pm 1\}$ it returns $\perp$; otherwise, it sets

$$\tau = \begin{cases} J_N(\gamma + 2r_{\mathsf{id}}) & \text{if } \sigma = 1 \\ J_N((\gamma + 2r_{\mathsf{id}})(d - 2r_{\mathsf{id}})(d - \gamma)) & \text{if } \sigma = -1 \end{cases}$$

and returns plaintext m as

$$m = \frac{1 - v \cdot \tau}{2}.$$

The value d used by DECRYPT is the smallest nonnegative integer d such that $J_N(d^2 - 4R_{\mathsf{id}}) = J_N(d^2 - 4uR_{\mathsf{id}}) = -1$. This value can also be precomputed.

Remark 8. As an alternative, the decryption algorithm can evaluate τ as $\tau = J_N(\gamma - 2r_{\mathsf{id}})$ when $\sigma = 1$ and as $\tau = J_N((\gamma - 2r_{\mathsf{id}})(d + 2r_{\mathsf{id}})(d - \gamma)) = -J_N((\gamma - 2r_{\mathsf{id}})(d_{\mathsf{id}} - 2r_{\mathsf{id}})(d_{\mathsf{id}} - \gamma)) = -J_N((\gamma + 2r_{\mathsf{id}})(d + 2r_{\mathsf{id}})(d - \gamma))$ when $\sigma = -1$. Also the value of $J_N(d + 2r_{\mathsf{id}})$ or of $J_N(d - 2r_{\mathsf{id}})$ can be precomputed.

Fourth exemplary embodiment

It is possible to *fix* the value of d as a global value to be used with all identities and to include it in the system parameters. We again illustrate the technique with the first exemplary embodiment. A similar embodiment can be obtained for the second exemplary embodiment.

This can be achieved by specializing hash function $\mathcal{H}$. Instead of considering a function mapping bit-strings to any element of $\mathbb{J}_N$, we require that, in addition, on input id, the output must satisfy the extra condition $J_N(d^2 - 4R_{\mathsf{id}}) = J_N(d^2 - 4uR_{\mathsf{id}}) = -1$ for some given d:

$$\begin{aligned} \mathcal{H}_d: \{0,1\}^* &\to \mathbb{J}_N, \mathsf{id} \\ &\mapsto \mathcal{H}_d(\mathsf{id}) \text{ satisfying } J_N(d^2 - 4\mathcal{H}_d(\mathsf{id})) \\ &= J_N(d^2 - 4u\mathcal{H}_d(\mathsf{id})) = -1. \end{aligned} \tag{2}$$

SETUP(1^κ) Given a security parameter κ, SETUP generates an RSA modulus $N = pq$ where p and q are prime. It also generates a random element $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$ and a global integer d. The public system parameters are $\mathsf{mpk} = \{N, u, d, \mathcal{H}_d\}$ where $\mathcal{H}_d$ is a cryptographic hash function as per Eq. (2). The master secret key is $\mathsf{msk} = \{p, q\}$.

$\text{EXTRACT}_{\text{msk}}(\text{id})$ Given identity id, key derivation algorithm EXTRACT first sets $R_{\text{id}} = \mathcal{H}_d(\text{id})$. If $R_{\text{id}} \in \mathbb{QR}_N$ it computes $r_{\text{id}} = {R_{\text{id}}}^{1/2} \mod N$; otherwise it computes $r_{\text{id}} = (uR_{\text{id}})^{1/2} \mod N$. EXTRACT returns user's private key $\text{usk} = \{r_{\text{id}}\}$.

$\text{ENCRYPT}_{\text{mpk}}(\text{id}, m)$ To encrypt a message $m \in \{0,1\}$ for a user with identity id, ENCRYPT first defines $R_{\text{id}} = \mathcal{H}_d(\text{id})$. It chooses at random $t, \bar{t} \in \mathbb{Z}/N\mathbb{Z}$ and lets

$$\varepsilon = (-1)^m J_N(t), \quad c^{(0)} = t + \frac{R_{id}}{t} \mod N, \quad c^{(1)} = \frac{c^{(0)}d + 4R_{id}}{c^{(0)} + d} \mod N,$$

$$\bar{\varepsilon} = (-1)^m J_N(\bar{t}), \quad \bar{c}^{(0)} = \bar{t} + \frac{uR_{id}}{\bar{t}} \mod N, \quad \bar{c}^{(1)} = \frac{\bar{c}^{(0)}d + 4uR_{id}}{\bar{c}^{(0)} + d} \mod N.$$

It chooses random bits $\beta_1, \beta_2 \in \{0,1\}$ and sets $c = c^{(\beta_1)}$ and $\bar{c} = \bar{c}^{(\beta_2)}$. The returned ciphertext is $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$.

$\text{DECRYPT}_{\text{usk}}(C)$ Let $R_{\text{id}} = \mathcal{H}_d(\text{id})$. From $\text{usk} = \{r_{\text{id}}\}$ and $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$, if ${r_{\text{id}}}^2 \equiv R_{\text{id}} \pmod N$, DECRYPT sets $\nu = \varepsilon$, $\gamma = c$ and $\Delta = R_{\text{id}}$; otherwise it sets $\nu = \bar{\varepsilon}$, $\gamma = \bar{c}$ and $\Delta = uR_{\text{id}}$. Next, it computes $\sigma = J_N(\gamma^2 - 4\Delta)$. If $\sigma \notin \{\pm 1\}$ it returns $\perp$; otherwise, it sets

$$\tau = \begin{cases} J_N(\gamma + 2r_{\text{id}}) & \text{if } \sigma = 1 \\ J_N((\gamma + 2r_{\text{id}})(d - 2r_{\text{id}})(d - \gamma)) & \text{if } \sigma = -1 \end{cases}$$

and returns plaintext m as

$$m = \frac{1 - \nu \cdot \tau}{2}.$$

Remark 9. In order to have short system parameters mpk, it may be advantageous to select a small value for d.

Remark 10. As an alternative, the decryption algorithm can evaluate τ as $\tau = J_N(\gamma - 2r_{\text{id}})$ when $\sigma = 1$ and as $\tau = J_N((\gamma - 2r_{\text{id}})(d + 2r_{\text{id}})(d - \gamma)) = -J_N((\gamma - 2r_{\text{id}})(d_{\text{id}} - 2r_{\text{id}})(d_{\text{id}} - \gamma)) = -J_N((\gamma + 2r_{\text{id}})(d + 2r_{\text{id}})(d - \gamma))$ when $\sigma = -1$. Also the value of $J_N(d + 2r_{\text{id}})$ or of $J_N(d - 2r_{\text{id}})$ can be precomputed.

Fifth exemplary embodiment

Assume now primes p and q satisfy the extra condition $p \equiv -q \pmod 4$. In this case, we know that $J_p(-1) = -J_q(-1)$ and therefore $J_N(-1) = -1$. Note that it is easily verified that N is the product of two primes that are inversely congruent modulo 4 by checking that $N \equiv 3 \pmod 4$.

This setting simplifies the cryptosystem. A nice observation is that $d = 0$ is a valid parameter when $N \equiv 3 \pmod 4$ since then $J_N(d^2 - 4\mathcal{H}(\mathsf{id})) = J_N(d^2 - 4u\mathcal{H}(\mathsf{id})) = J_N(-1) = -1$ as desired. Any cryptographic hash function $\mathcal{H}$ mapping bit-strings to elements of $\mathbb{J}_N$ can be used.

SETUP(1^κ) Given a security parameter κ, SETUP generates an RSA modulus $N = pq$ where p and q are prime and $p \equiv -q \pmod 4$. It also generates a random element $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$. The public system parameters are $\mathsf{mpk} = \{N, u, \mathcal{H}\}$ where $\mathcal{H}$ is a cryptographic hash function mapping bit-strings to elements of $\mathbb{J}_N$. The master secret key is $\mathsf{msk} = \{p, q\}$.

EXTRACT$_{\mathsf{msk}}(\mathsf{id})$ Given identity id, key derivation algorithm EXTRACT first sets $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$. If $R_{\mathsf{id}} \in \mathbb{QR}_N$ it computes $r_{\mathsf{id}} = {R_{\mathsf{id}}}^{1/2} \bmod N$; otherwise it computes $r_{\mathsf{id}} = (uR_{\mathsf{id}})^{1/2} \bmod N$. EXTRACT returns user's private key $\mathsf{usk} = \{r_{\mathsf{id}}\}$.

ENCRYPT$_{\mathsf{mpk}}(\mathsf{id}, m)$ To encrypt a message $m \in \{0,1\}$ for a user with identity id, ENCRYPT first defines $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$. It chooses at random $t, \bar{t} \in \mathbb{Z}/N\mathbb{Z}$ and lets

$$\varepsilon = (-1)^m J_N(t), \quad c^{(0)} = t + \frac{R_{id}}{t} \bmod N, \quad c^{(1)} = \frac{4R_{id}}{c^{(0)}} \bmod N,$$

$$\bar{\varepsilon} = (-1)^m J_N(\bar{t}), \quad \bar{c}^{(0)} = \bar{t} + \frac{uR_{id}}{\bar{t}} \bmod N, \quad \bar{c}^{(1)} = \frac{4uR_{id}}{\bar{c}^{(0)}} \bmod N.$$

It chooses random bits $\beta_1, \beta_2 \in \{0,1\}$ and sets $c = c^{(\beta_1)}$ and $\bar{c} = \bar{c}^{(\beta_2)}$. The returned ciphertext is $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$.

DECRYPT$_{\mathsf{usk}}(C)$ Let $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$. From $\mathsf{usk} = \{r_{\mathsf{id}}\}$ and $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$, if ${r_{\mathsf{id}}}^2 \equiv R_{\mathsf{id}} \pmod N$, DECRYPT sets $\nu = \varepsilon$, $\gamma = c$ and $\Delta = R_{\mathsf{id}}$; otherwise it sets $\nu = \bar{\varepsilon}$, $\gamma = \bar{c}$ and $\Delta = uR_{\mathsf{id}}$. Next, it computes $\sigma = J_N(\gamma^2 - 4\Delta)$. If $\sigma \notin \{\pm 1\}$ it returns $\perp$; otherwise, it sets

$$\tau = \begin{cases} J_N(\gamma + 2r_{\mathsf{id}}) & \text{if } \sigma = 1 \\ J_N((\gamma + 2r_{\mathsf{id}})(2r_{\mathsf{id}}\gamma)) & \text{if } \sigma = -1 \end{cases}$$

and returns plaintext m as

$$m = \frac{1 - \nu \cdot \tau}{2}.$$

Remark 11. For better efficiency, as a variant, the encryption algorithm can first choose β_1, β_2 at random in $\{0,1\}$ and then sets $c = t + \frac{R_{id}}{t} \bmod N$ if $\beta_1 = 0$ and

$c = \frac{4R_{id}t}{t^2+R_{id}} \mod N$ if $\beta_1 = 1$; and $\bar{c} = \bar{t} + \frac{uR_{id}}{\bar{t}} \mod N$ if $\beta_2 = 0$ and $\bar{c} = \frac{4uR_{id}\bar{t}}{\bar{t}^2+uR_{id}} \mod N$ if $\beta_2 = 1$.

Remark 12. As an alternative, the decryption algorithm can evaluate τ as $\tau = J_N(\gamma - 2r_{\mathrm{id}})$ when $\sigma = 1$ and as $\tau = -J_N((\gamma - 2r_{\mathrm{id}})(2r_{\mathrm{id}}\gamma))$ when $\sigma = -1$. Also the value of $J_N(2r_{\mathrm{id}})$ can be precomputed.

Sixth exemplary embodiment

The next embodiment corresponds to the case $N \equiv 3 \pmod 4$ and $d = 0$ applied to the second exemplary embodiment.

Assume $R_{\mathrm{id}} = \mathcal{H}(\mathrm{id}) \in \mathbb{QR}_N$. Then, using the notation of the second exemplary embodiment, we notice that $J_N(-(td + 2R_{\mathrm{id}})(td + 2t^2)) = J_N(-4R_{\mathrm{id}}t^2) = J_N(-1) = -1$ when $d = 0$. Hence, in the encryption process, if $c \neq c'$ we always have $\varepsilon = \varepsilon' \cdot J_N(-1) = -\bar{\varepsilon}$, whatever the value of β_1. The case $uR_{\mathrm{id}} = u\mathcal{H}(\mathrm{id}) \in \mathbb{QR}_N$ (i.e., $\mathcal{H}(\mathrm{id}) \in \mathbb{J}_N \setminus \mathbb{QR}_N$) is similar. We always have $\bar{\varepsilon} = \bar{\varepsilon}' \cdot J_N(-1) = -\bar{\varepsilon}$, whatever the value of β_2.

SETUP(1^κ) Given a security parameter κ, SETUP generates an RSA modulus $N = pq$ where p and q are prime and $p \equiv -q \pmod 4$. It also generates a random element $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$. The public system parameters are $\mathsf{mpk} = \{N, u, \mathcal{H}\}$ where $\mathcal{H}$ is a cryptographic hash function mapping bit-strings to elements of $\mathbb{J}_N$. The master secret key is $\mathsf{msk} = \{p, q\}$.

EXTRACT$_{\mathsf{msk}}(\mathsf{id})$ Given identity id, key derivation algorithm EXTRACT first sets $R_{\mathrm{id}} = \mathcal{H}(\mathrm{id})$. If $R_{\mathrm{id}} \in \mathbb{QR}_N$ it computes $r_{\mathrm{id}} = {R_{\mathrm{id}}}^{1/2} \mod N$; otherwise it computes $r_{\mathrm{id}} = (uR_{\mathrm{id}})^{1/2} \mod N$. EXTRACT returns user's private key $\mathsf{usk} = \{r_{\mathrm{id}}\}$.

ENCRYPT$_{\mathsf{mpk}}(\mathsf{id}, m)$ To encrypt a message $m \in \{0,1\}$ for a user with identity id, ENCRYPT first defines $R_{\mathrm{id}} = \mathcal{H}(\mathrm{id})$. It chooses at random $t, \bar{t} \in \mathbb{Z}/N\mathbb{Z}$ and lets

$$\varepsilon' = (-1)^m J_N(t), \quad c'^{(0)} = t + \frac{R_{id}}{t} \mod N, \quad c'^{(1)} = \frac{4R_{id}}{c'^{(0)}} \mod N,$$

$$\bar{\varepsilon}' = (-1)^m J_N(\bar{t}), \quad \bar{c}'^{(0)} = \bar{t} + \frac{uR_{id}}{\bar{t}} \mod N, \quad \bar{c}'^{(1)} = \frac{4uR_{id}}{\bar{c}'^{(0)}} \mod N.$$

It chooses random bits $\beta_1, \beta_2 \in \{0,1\}$ and sets $c' = c'^{(\beta_1)}$ and $\bar{c}' = \bar{c}'^{(\beta_2)}$. Define $c = \min(c', N - c')$ and $\bar{c} = \min(\bar{c}', N - \bar{c}')$. If $c = c'$ then define $\varepsilon = \varepsilon'$;

otherwise define $\varepsilon = -\varepsilon'$. Likewise, if $\bar{c} = \bar{c}'$ then define $\bar{\varepsilon} = \bar{\varepsilon}'$; otherwise define $\bar{\varepsilon} = -\bar{\varepsilon}'$. The returned ciphertext is $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$.

$\text{DECRYPT}_{\mathsf{usk}}(C)$ Let $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$. From $\mathsf{usk} = \{r_{\mathsf{id}}\}$ and $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$, if ${r_{\mathsf{id}}}^2 \equiv R_{\mathsf{id}} \pmod{N}$, DECRYPT sets $\nu = \varepsilon$, $\gamma = c$ and $\Delta = R_{\mathsf{id}}$; otherwise it sets $\nu = \bar{\varepsilon}$, $\gamma = \bar{c}$ and $\Delta = uR_{\mathsf{id}}$. Next, it computes $\sigma = J_N(\gamma^2 - 4\Delta)$. If $\sigma \notin \{\pm 1\}$ it returns $\perp$; otherwise, from $d = \mathcal{G}_{\mathsf{id}}(s)$, it sets

$$\tau = \begin{cases} J_N(\gamma + 2r_{\mathsf{id}}) & \text{if } \sigma = 1 \\ J_N((\gamma + 2r_{\mathsf{id}})(2r_{\mathsf{id}}\gamma)) & \text{if } \sigma = -1 \end{cases}$$

and returns plaintext m as

$$m = \frac{1 - \nu \cdot \tau}{2}.$$

Remark 13. For better efficiency, as a variant, the encryption algorithm can first choose β_1, β_2 at random in $\{0,1\}$ and then sets $c' = t + \frac{R_{id}}{t} \mod N$ if $\beta_1 = 0$ and $c' = \frac{4R_{id}t}{t^2 + R_{id}} \mod N$ if $\beta_1 = 1$; and $\bar{c}' = \bar{t} + \frac{uR_{id}}{\bar{t}} \mod N$ if $\beta_2 = 0$ and $\bar{c}' = \frac{4uR_{id}\bar{t}}{\bar{t}^2 + uR_{id}} \mod N$ if $\beta_2 = 1$.

Remark 14. As an alternative, the decryption algorithm can evaluate τ as $\tau = J_N(\gamma - 2r_{\mathsf{id}})$ when $\sigma = 1$ and as $\tau = -J_N((\gamma - 2r_{\mathsf{id}})(2r_{\mathsf{id}}\gamma))$ when $\sigma = -1$. Also the value of $J_N(2r_{\mathsf{id}})$ can be precomputed.

There are several advantages for the proposed cryptosystems:

- identity-based paradigm;
- both confidentiality *and* anonymity;
- short ciphertexts;
- fast encryption/decryption times;
- building blocks for PEKS (public-key encrypton with keyword search);
- strong security guarantees.

The proposed encryption systems can be used in any application requiring efficient *anonymous* identity-based encryption. Compared with [14], the new cryptosystems proposed in this application present the advantage of being identity-based. This solves a number of practical issues in real systems. However, the ciphertexts are twice longer.

An important application of anonymous IBEs is *public key encryption with keyword search* (PEKS) [7]. PEKS allows searching on data that is encrypted using a public-key system. A typical application is for an email gateway to test whether or not the keyword "urgent" is present in an email. The gateway then routes the email if it is the case. Of course the gateway should only learn whether the word "urgent" is present but nothing else about the email. Further applications for PEKS can be found in [7].

FIG. 1 illustrates a block diagram of an exemplary system in which various aspects of the exemplary embodiments of the present principles may be implemented. System 100 may be embodied as a device including the various components described below and is configured to perform the processes described above. Examples of such devices, include, but are not limited to, personal computers, laptop computers, smartphones, tablet computers, digital multimedia set top boxes, digital television receivers, personal video recording systems, connected home appliances, and servers. System 100 may be communicatively coupled to other similar systems, and to trusted third parties via a communication channel as shown in Figure 2 and as known by those skilled in the art to implement the exemplary cryptosystems described above.

The system 100 may include at least one processor 110 configured to execute instructions loaded therein for implementing the various processes as discussed above. Processor 110 may include embedded memory, input output interface and various other circuitries as known in the art. The system 100 may also include at least one memory 120 (e.g., a volatile memory device, a non-volatile memory device). System 100 may additionally include a storage device 140, which may include non-volatile memory, including, but not limited to, EEPROM, ROM, PROM, RAM, DRAM, SRAM, flash, magnetic disk drive, and/or optical disk drive. The storage device 140 may comprise an internal storage device, an attached storage device and/or a network accessible storage device, as non-limiting examples. System 100 may also include an encryption/decryption module 130 configured to process data to provide an encrypted message or decrypted message.

Encryption/decryption module 130 represents the module(s) that may be included in a device to perform the encryption and/or decryption functions. As is known, a device may include one or both of the encryption and decryption modules, for example, encryption may be done on a regular PC since encryption does not involve secret key so that the PC need not include secure memory for storing the input

parameters (i.e., the public system parameters and the user's identity). Decryption however, requires secret keys (i.e., the decryption key) and is done in a secure device, for example a smart card. As memory is expensive on smart card, the encryption functionality may not always be provided on a smart card. The encryption and/or decryption may be performed using shared resources as known to those skilled in the art. Additionally, encryption/decryption module 130 may be implemented as a separate element of system 100 or may be incorporated within processors 110 as a combination of hardware and software as known to those skilled in the art.

Program code to be loaded onto processors 110 to perform the various processes described hereinabove may be stored in storage device 140 and subsequently loaded onto memory 120 for execution by processors 110. In accordance with the exemplary embodiments of the present principles, one or more of the processor(s) 110, memory 120, storage device 140 and encryption/decryption module 130 may store one or more of the various items during the performance of the processes discussed herein above, including, but not limited to a public system parameters, a private key, encrypted messages, equations, formula, matrices, variables, operations, and operational logic.

The system 100 may also include communication interface 150 that enables communication with other devices via communication channel 160. The communication interface 150 may include, but is not limited to a transceiver configured to transmit and receive data from communication channel 160. The communication interface may include, but is not limited to, a modem or network card and the communication channel may be implemented within a wired and/or wireless medium. The various components of system 100 may be connected or communicatively coupled together using various suitable connections, including, but not limited to internal buses, wires, and printed circuit boards.

As a non-limiting example, one or more of the above-identified components may receive and/or store the information (e.g., to be encrypted) and/or the ciphertext (e.g., to be decrypted, to be operated on homomorphically, resulting from encryption). As a further non-limiting example, one or more of the above-identified components may receive and/or store the encryption function(s) and/or the decryption function(s), as described herein above.

The exemplary embodiments of this invention may be carried out by computer software implemented by the processor 110 or by hardware, or by a combination of

hardware and software. As a non-limiting example, the exemplary embodiments of this invention may be implemented by one or more integrated circuits. The memory 120 may be of any type appropriate to the technical environment and may be implemented using any appropriate data storage technology, such as optical memory devices, magnetic memory devices, semiconductor-based memory devices, fixed memory and removable memory, as non-limiting examples. The processor 110 may be of any type appropriate to the technical environment, and may encompass one or more of microprocessors, general purpose computers, special purpose computers and processors based on a multi-core architecture, as non-limiting examples.

Figure 2 illustrates an arrangement wherein data is exchanged between two terminals 210 and 220 in accordance with the present principles. Each of the terminals 210 and 220 include encryptor/decryptor modules 230 and 240, respectively, and may additionally include each of the other components of system 100 described above, as appropriate. Terminals 210 and 220 are communicatively coupled to each other via communication channel 250, which may be implemented via wired and/or wireless medium. Additionally, arrangement 200 may include a trusted third party 260 communicatively coupled to terminals 210 and 220, wherein third party 260 may in some cases, among other things, generate common parameters and the keys, distribute them to the terminals, and/or generate common keys in a manner known to those skilled in the art. For example, in identity based encryption, the setup algorithm is performed by the trusted third party to generate, among other things, the master secret key. Following key generation, a message can be encrypted and decrypted by the terminals as described above, and transmitted and received via communication channel 250.

Figure 3 illustrates a generalized flow diagram of an identity-based cryptosystem.

Figures 4 to 6 show exemplary flow charts according to the present principles. The flow charts illustrate the anonymous identity-based crypto processes discussed above. The processes of Figures 4 - 6 may be executed by e.g., a processor 110 of Figure 1. The processes may represent, e.g., computer program products having the computer-executable instructions which may be stored in non-transitory computer-readable storage media 120 of Figure 1 as described before.

The foregoing has provided by way of exemplary embodiments and non-limiting examples a description of the method and systems contemplated by the

inventor. It is clear that various modifications and adaptations may become apparent to those skilled in the art in view of the description. However, such various modifications and adaptations fall within the scope of the teachings of the various embodiments described above.

The embodiments described herein may be implemented in, for example, a method or a process, an apparatus, a software program, a data stream, or a signal. Even if only discussed in the context of a single form of implementation (for example, discussed only as a method), the implementation of features discussed above may also be implemented in other forms (for example, an apparatus or program). An apparatus may be implemented in, for example, appropriate hardware, software, and firmware. The methods may be implemented in, for example, an apparatus such as, for example, a processor, which refers to processing devices in general, including, for example, a computer, a microprocessor, an integrated circuit, or a programmable logic device. Processors also include communication devices, such as, for example, computers, cell phones, portable/personal digital assistants ("PDAs"), and other devices that facilitate communication of information between end-users.

Reference to “one embodiment” or “an embodiment” or “one implementation” or “an implementation” of the present principles, as well as other variations thereof, mean that a particular feature, structure, characteristic, and so forth described in connection with the embodiment is included in at least one embodiment of the present principles. Thus, the appearances of the phrase “in one embodiment” or “in an embodiment” or “in one implementation” or “in an implementation”, as well any other variations, appearing in various places throughout the specification are not necessarily all referring to the same embodiment.

Additionally, this application or its claims may refer to “determining” various pieces of information. Determining the information may include one or more of, for example, estimating the information, calculating the information, predicting the information, or retrieving the information from memory.

Further, this application or its claims may refer to “accessing” various pieces of information. Accessing the information may include one or more of, for example, receiving the information, retrieving the information (for example, from memory), storing the information, processing the information, transmitting the information, moving the information, copying the information, erasing the information, calculating

the information, determining the information, predicting the information, or estimating the information.

Additionally, this application or its claims may refer to "receiving" various pieces of information. Receiving is, as with "accessing", intended to be a broad term. Receiving the information may include one or more of, for example, accessing the information, or retrieving the information (for example, from memory). Further, "receiving" is typically involved, in one way or another, during operations such as, for example, storing the information, processing the information, transmitting the information, moving the information, copying the information, erasing the information, calculating the information, determining the information, predicting the information, or estimating the information.

As will be evident to one of skill in the art, implementations may produce a variety of signals formatted to carry information that may be, for example, stored or transmitted. The information may include, for example, instructions for performing a method, or data produced by one of the described embodiments. For example, a signal may be formatted to carry the bitstream of a described embodiment. Such a signal may be formatted, for example, as an electromagnetic wave (for example, using a radio frequency portion of spectrum) or as a baseband signal. The formatting may include, for example, encoding a data stream and modulating a carrier with the encoded data stream. The information that the signal carries may be, for example, analog or digital information. The signal may be transmitted over a variety of different wired and/or wireless links, as is known. The signal may be stored on a processor-readable medium.

While several embodiments have been described and illustrated herein, those of ordinary skill in the art will readily envision a variety of other means and/or structures for performing the functions and/or obtaining the results and/or one or more of the advantages described herein, and each of such variations and/or modifications is deemed to be within the scope of the present embodiments. More generally, those skilled in the art will readily appreciate that all parameters, dimensions, materials, and configurations described herein are meant to be exemplary and that the actual parameters, dimensions, materials, and/or configurations will depend upon the specific application or applications for which the teachings herein is/are used. Those skilled in the art will recognize, or be able to ascertain using no more than routine experimentation, many equivalents to the specific embodiments described herein. It

is, therefore, to be understood that the foregoing embodiments are presented by way of example only and that, within the scope of the appended claims and equivalents thereof, the embodiments disclosed may be practiced otherwise than as specifically described and claimed. The present embodiments are directed to each individual feature, system, article, material and/or method described herein. In addition, any combination of two or more such features, systems, articles, materials and/or methods, if such features, systems, articles, materials and/or methods are not mutually inconsistent, is included within the scope of the present embodiment.

CLAIMS

1. A method for communicating a message m, comprising:

accessing the message m;

accessing a user identity id and public system parameters $\mathsf{mpk} = \{N, u, \mathcal{H}, \{\mathcal{G}_{\mathsf{id}}\}, s\}$, where N is a composite integer, $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$, s is a bit-string, $\mathcal{H}$ is a cryptographic hash function mapping bit-strings to elements of $\mathbb{J}_N$ and $\{\mathcal{G}_{\mathsf{id}}\}$ is a family of functions mapping bit-strings to elements of a subset of $\mathbb{Z}/N\mathbb{Z}$, namely $\mathcal{G}_{\mathsf{id}}: \{0,1\}^* \to \mathbb{Z}/N\mathbb{Z}, s \mapsto d = \mathcal{G}_{\mathsf{id}}(s)$ such that $J_N(d^2 - 4\mathcal{H}(\mathsf{id})) = J_N(d^2 - 4u\mathcal{H}(\mathsf{id})) = -1$;

generating a ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ of the message m using the user identity id, where

$$\varepsilon = (-1)^m J_N(t), \quad c^{(0)} = t + \frac{R_{id}}{t} \bmod N, \quad c^{(1)} = \frac{c^{(0)} d + 4R_{id}}{c^{(0)} + d} \bmod N,$$

$$\bar{\varepsilon} = (-1)^m J_N(\bar{t}), \quad \bar{c}^{(0)} = \bar{t} + \frac{uR_{id}}{\bar{t}} \bmod N, \quad \bar{c}^{(1)} = \frac{\bar{c}^{(0)} d + 4uR_{id}}{\bar{c}^{(0)} + d} \bmod N,$$

where $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$, $t, \bar{t} \in (\mathbb{Z}/N\mathbb{Z})^\times$, $d = \mathcal{G}_{\mathsf{id}}(s)$, choosing random bits $\beta_1, \beta_2 \in \{0,1\}$ and setting $c = c^{(\beta_1)}$ and $\bar{c} = \bar{c}^{(\beta_2)}$; and

transmitting the ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ via a communication channel.

2. The method of claim 1, wherein $N = pq$, p and q are prime.

3. The method of claim 1, wherein the message $m \in \{0,1\}$.

4. The method of claim 1, wherein the ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ is determined instead by the following steps:

setting

$$\varepsilon' = (-1)^m J_N(t), \quad c'^{(0)} = t + \frac{R_{id}}{t} \bmod N, \quad c'^{(1)} = \frac{c'^{(0)} d + 4R_{id}}{c'^{(0)} + d} \bmod N,$$

$$\bar{\varepsilon}' = (-1)^m J_N(\bar{t}), \quad \bar{c}'^{(0)} = \bar{t} + \frac{uR_{id}}{\bar{t}} \bmod N, \quad \bar{c}'^{(1)} = \frac{\bar{c}'^{(0)} d + 4uR_{id}}{\bar{c}'^{(0)} + d} \bmod N;$$

choosing random bits $\beta_1, \beta_2 \in \{0,1\}$;

setting $c' = c'^{(\beta_1)}$ and $\bar{c}' = \bar{c}'^{(\beta_2)}$;

setting $c = \min(c', N - c')$ and $\bar{c} = \min(\bar{c}', N - \bar{c}')$;

if $c = c'$ then setting $\varepsilon = \varepsilon'$; otherwise setting

$$\varepsilon = \begin{cases} \varepsilon' \cdot J_N(-1) & \text{if } \beta_1 = 0 \\ \varepsilon' \cdot J_N(-(td + 2R_{\mathrm{id}})(td + 2t^2)) & \text{if } \beta_1 = 1 \end{cases}; \text{ and}$$

if $\bar{c} = \bar{c}'$ then setting $\bar{\varepsilon} = \bar{\varepsilon}'$; otherwise setting

$$\bar{\varepsilon} = \begin{cases} \bar{\varepsilon}' \cdot J_N(-1) & \text{if } \beta_2 = 0 \\ \bar{\varepsilon}' \cdot J_N(-(\bar{t}d + 2uR_{\mathrm{id}})(\bar{t}d + 2\bar{t}^2)) & \text{if } \beta_2 = 1 \end{cases}$$

5. The method of claim 1, wherein d is the smallest nonnegative integer d such that such that $J_N(d^2 - 4\mathcal{H}(\mathrm{id})) = J_N(d^2 - 4u\mathcal{H}(\mathrm{id})) = -1$.

6. The method of claim 1, wherein d is fixed as a global value to be used with all identities.

7. The method of claim 1, wherein N is such that $J_N(-1) = -1$.

8. The method of claim 7, wherein $d = 0$.

9. A method for processing a ciphertext, comprising:

receiving the ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ for a user with an identity id via a communication channel;

accessing the ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ of a plaintext message m, where

$$\varepsilon = (-1)^m J_N(t), \quad c^{(0)} = t + \frac{R_{id}}{t} \bmod N, \quad c^{(1)} = \frac{c^{(0)}d + 4R_{id}}{c^{(0)} + d} \bmod N,$$

$$\bar{\varepsilon} = (-1)^m J_N(\bar{t}), \quad \bar{c}^{(0)} = \bar{t} + \frac{uR_{id}}{\bar{t}} \bmod N, \quad \bar{c}^{(1)} = \frac{\bar{c}^{(0)}d + 4uR_{id}}{\bar{c}^{(0)} + d} \bmod N,$$

where N is a composite integer, $R_{\mathrm{id}} = \mathcal{H}(\mathrm{id})$, $t, \bar{t} \in (\mathbb{Z}/N\mathbb{Z})^{\times}$, $d = \mathcal{G}_{\mathrm{id}}(s)$, $c = c^{(\beta_1)}$ and $\bar{c} = \bar{c}^{(\beta_2)}$ for random bits $\beta_1, \beta_2 \in \{0,1\}$, wherein public system parameters $\mathsf{mpk} = \{N, u, \mathcal{H}, \{\mathcal{G}_{\mathrm{id}}\}, s\}$, where N is a composite integer, $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$, s is bit-string, $\mathcal{H}$ is a cryptographic hash function mapping bit-strings to elements of $\mathbb{J}_N$ and $\{\mathcal{G}_{\mathrm{id}}\}$ is a family of functions mapping bit-strings to elements of a subset of $\mathbb{Z}/N\mathbb{Z}$, namely $\mathcal{G}_{\mathrm{id}}: \{0,1\}^* \to \mathbb{Z}/N\mathbb{Z}, s \mapsto d = \mathcal{G}_{\mathrm{id}}(s)$ such that $J_N(d^2 - 4\mathcal{H}(\mathrm{id})) = J_N(d^2 - 4u\mathcal{H}(\mathrm{id})) = -1$; and

generating the message m by accessing a private key $\mathsf{usk} = \{r_{\mathrm{id}}\}$ for a user with an identity id by following steps:

if ${r_{\mathrm{id}}}^2 \equiv R_{\mathrm{id}} \pmod N$, setting $\nu = \varepsilon$, $\gamma = c$ and $\Delta = R_{\mathrm{id}}$; otherwise setting $\nu = \bar{\varepsilon}$, $\gamma = \bar{c}$ and $\Delta = uR_{\mathrm{id}}$;

determining $\sigma = J_N(\gamma^2 - 4\Delta)$,

setting

$$\tau = \begin{cases} J_N(\gamma + 2r_{\mathrm{id}}) & \text{if } \sigma = 1 \\ J_N((\gamma + 2r_{\mathrm{id}})(d - 2r_{\mathrm{id}})(d - \gamma)) & \text{if } \sigma = -1 \end{cases}, \text{ where } d = \mathcal{G}_{\mathrm{id}}(s); \text{ and}$$

determining the message m as

$$m = \frac{1 - \nu \cdot \tau}{2}.$$

10. The method of claim 9, wherein the private key $\mathsf{usk} = \{r_{\mathrm{id}}\}$ for the user with the identity id is generated by a master secrete key msk, wherein $R_{\mathrm{id}} = \mathcal{H}(\mathrm{id})$, and if $R_{\mathrm{id}} \in \mathbb{QR}_N$ then ${r_{\mathrm{id}}} = {R_{\mathrm{id}}}^{1/2} \bmod N$; otherwise, $r_{\mathrm{id}} = (uR_{\mathrm{id}})^{1/2} \bmod N$.

11. The method of claim 10, wherein $N = pq$, p and q are prime.

12. The method of claim 11, wherein the master secret key is $\mathsf{msk} = \{p, q\}$.

13. The method of claim 9, wherein the message $m \in \{0,1\}$.

14. The method of claim 9, wherein d is the smallest nonnegative integer d such that such that $J_N(d^2 - 4\mathcal{H}(\mathsf{id})) = J_N(d^2 - 4u\mathcal{H}(\mathsf{id})) = -1$.

15. The method of claim 9, wherein d is fixed as a global value to be used with all identities.

16. The method of claim 9, wherein N is such that $J_N(-1) = -1$.

17. The method of claim 16, wherein $d = 0$.

18. A method for generating keys for an identity-based cryptosystem wherein a private key is used to decrypt ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ of a plaintext message m for a user with an identity id where

$$\varepsilon = (-1)^m J_N(t), \quad c^{(0)} = t + \frac{R_{id}}{t} \bmod N, \quad c^{(1)} = \frac{c^{(0)}d + 4R_{id}}{c^{(0)} + d} \bmod N,$$

$$\bar{\varepsilon} = (-1)^m J_N(\bar{t}), \quad \bar{c}^{(0)} = \bar{t} + \frac{uR_{id}}{\bar{t}} \bmod N, \quad \bar{c}^{(1)} = \frac{\bar{c}^{(0)}d + 4uR_{id}}{\bar{c}^{(0)} + d} \bmod N,$$

where N is a composite integer, $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$, $t, \bar{t} \in (\mathbb{Z}/N\mathbb{Z})^{\times}$, $d = \mathcal{G}_{\mathsf{id}}(s)$, $c = c^{(\beta_1)}$ and $\bar{c} = \bar{c}^{(\beta_2)}$ for random bits $\beta_1, \beta_2 \in \{0,1\}$, comprising:

setting a master secret key msk; and

generating, using the master secret key msk, the private key $\mathsf{usk} = \{r_{\mathsf{id}}\}$ for a user with an identity id, using public system parameters $\mathsf{mpk} = \{N, u, \mathcal{H}, \{\mathcal{G}_{\mathsf{id}}\}, s\}$, where N is a composite integer, $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$, s is bit-string, $\mathcal{H}$ is a cryptographic hash function mapping bit-strings to elements of $\mathbb{J}_N$ and $\{\mathcal{G}_{\mathsf{id}}\}$ is a family of functions mapping bit-strings to elements of a subset of $\mathbb{Z}/N\mathbb{Z}$ namely $\mathcal{G}_{\mathsf{id}}: \{0,1\}^* \to \mathbb{Z}/N\mathbb{Z}, s \mapsto d = \mathcal{G}_{\mathsf{id}}(s)$ such that $J_N(d^2 - 4\mathcal{H}(\mathsf{id})) = J_N(d^2 - 4u\mathcal{H}(\mathsf{id})) = -1$.

19. The method of claim 18, wherein $N = pq$, p and q are prime.

20. The method of claim 19, wherein $\mathsf{msk} = \{p, q\}$.

21. The method of claim 18, wherein N is such that $J_N(-1) = -1$.

22. An apparatus for communicating a message m, comprising:

a processor configured to access the message m, the processor further configured to access a user identity id and public system parameters $\mathsf{mpk} = \{N, u, \mathcal{H}, \{\mathcal{G}_{\mathsf{id}}\}, s\}$, where N is a composite integer, $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$, s is a bit-string, $\mathcal{H}$ is a cryptographic hash function mapping bit-strings to elements of $\mathbb{J}_N$ and $\{\mathcal{G}_{\mathsf{id}}\}$ is a family of functions mapping bit-strings to elements of a subset of $\mathbb{Z}/N\mathbb{Z}$, namely $\mathcal{G}_{\mathsf{id}}: \{0,1\}^* \to \mathbb{Z}/N\mathbb{Z}, s \mapsto d = \mathcal{G}_{\mathsf{id}}(s)$ such that $J_N(d^2 - 4\mathcal{H}(\mathsf{id})) = J_N(d^2 - 4u\mathcal{H}(\mathsf{id})) = -1$;

an encryptor configured to generate a ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ of the message m using the user identity id, where

$$\varepsilon = (-1)^m J_N(t), \quad c^{(0)} = t + \frac{R_{id}}{t} \bmod N, \quad c^{(1)} = \frac{c^{(0)} d + 4R_{id}}{c^{(0)} + d} \bmod N,$$

$$\bar{\varepsilon} = (-1)^m J_N(\bar{t}), \quad \bar{c}^{(0)} = \bar{t} + \frac{uR_{id}}{\bar{t}} \bmod N, \quad \bar{c}^{(1)} = \frac{\bar{c}^{(0)} d + 4uR_{id}}{\bar{c}^{(0)} + d} \bmod N,$$

where $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$, $t, \bar{t} \in (\mathbb{Z}/N\mathbb{Z})^\times$, $d = \mathcal{G}_{\mathsf{id}}(s)$, choosing random bits $\beta_1, \beta_2 \in \{0,1\}$ and setting $c = c^{(\beta_1)}$ and $\bar{c} = \bar{c}^{(\beta_2)}$; and

a communication interface, coupled to a communication channel, configured to transmit the ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ via the communication channel.

23. The apparatus of claim 22, wherein $N = pq$, p and q are prime.

24. The apparatus of claim 22, wherein the message $m \in \{0,1\}$.

25. The apparatus of claim 23, wherein the ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ is determined instead by the following steps:

setting

$$\varepsilon' = (-1)^m J_N(t), \quad c'^{(0)} = t + \frac{R_{id}}{t} \bmod N, \quad c'^{(1)} = \frac{c'^{(0)} d + 4R_{id}}{c'^{(0)} + d} \bmod N,$$

$$\bar{\varepsilon}' = (-1)^m J_N(\bar{t}), \quad \bar{c}'^{(0)} = \bar{t} + \frac{uR_{id}}{\bar{t}} \bmod N, \quad \bar{c}'^{(1)} = \frac{\bar{c}'^{(0)} d + 4uR_{id}}{\bar{c}'^{(0)} + d} \bmod N;$$

choosing random bits $\beta_1, \beta_2 \in \{0,1\}$;

setting $c' = c'^{(\beta_1)}$ and $\bar{c}' = \bar{c}'^{(\beta_2)}$;

setting $c = \min(c', N - c')$ and $\bar{c} = \min(\bar{c}', N - \bar{c}')$;

if $c = c'$ then setting $\varepsilon = \varepsilon'$; otherwise setting

$$\varepsilon = \begin{cases} \varepsilon' \cdot J_N(-1) & \text{if } \beta_1 = 0 \\ \varepsilon' \cdot J_N(-(td + 2R_{\mathrm{id}})(td + 2t^2)) & \text{if } \beta_1 = 1 \end{cases}; \text{ and}$$

if $\bar{c} = \bar{c}'$ then setting $\bar{\varepsilon} = \bar{\varepsilon}'$; otherwise setting

$$\bar{\varepsilon} = \begin{cases} \bar{\varepsilon}' \cdot J_N(-1) & \text{if } \beta_2 = 0 \\ \bar{\varepsilon}' \cdot J_N(-(\bar{t}d + 2uR_{\mathrm{id}})(\bar{t}d + 2\bar{t}^2)) & \text{if } \beta_2 = 1 \end{cases}.$$

26. The apparatus of claim 22, wherein d is the smallest nonnegative integer d such that such that $J_N(d^2 - 4\mathcal{H}(\mathrm{id})) = J_N(d^2 - 4u\mathcal{H}(\mathrm{id})) = -1$.

27. The apparatus of claim 22, wherein d is fixed as a global value to be used with all identities.

28. The apparatus of claim 22, wherein N is such that $J_N(-1) = -1$.

29. The apparatus of claim 28, wherein $d = 0$.

30. An apparatus for processing a ciphertext, comprising:

a communication interface, coupled to a communication channel, configured to receive the ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ of a plaintext message m, where

$$\varepsilon = (-1)^m J_N(t), \quad c^{(0)} = t + \frac{R_{id}}{t} \bmod N, \quad c^{(1)} = \frac{c^{(0)} d + 4R_{id}}{c^{(0)} + d} \bmod N,$$

$$\bar{\varepsilon} = (-1)^m J_N(\bar{t}), \quad \bar{c}^{(0)} = \bar{t} + \frac{uR_{id}}{\bar{t}} \bmod N, \quad \bar{c}^{(1)} = \frac{\bar{c}^{(0)} d + 4uR_{id}}{\bar{c}^{(0)} + d} \bmod N,$$

where N is a composite integer, $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$, $t, \bar{t} \in (\mathbb{Z}/N\mathbb{Z})^{\times}$, $d = \mathcal{G}_{\mathsf{id}}(s)$, $c = c^{(\beta_1)}$ and $\bar{c} = \bar{c}^{(\beta_2)}$ for random bits $\beta_1, \beta_2 \in \{0,1\}$, wherein public system parameters $\mathsf{mpk} = \{N, u, \mathcal{H}, \{\mathcal{G}_{\mathsf{id}}\}, s\}$, where N is a composite integer, $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$, s is bit-string, $\mathcal{H}$ is a cryptographic hash function mapping bit-strings to elements of $\mathbb{J}_N$ and $\{\mathcal{G}_{\mathsf{id}}\}$ is a family of functions mapping bit-strings to elements of a subset of $\mathbb{Z}/N\mathbb{Z}$, namely $\mathcal{G}_{\mathsf{id}}: \{0,1\}^* \to \mathbb{Z}/N\mathbb{Z}, s \mapsto d = \mathcal{G}_{\mathsf{id}}(s)$ such that $J_N(d^2 - 4\mathcal{H}(\mathsf{id})) = J_N(d^2 - 4u\mathcal{H}(\mathsf{id})) = -1$; and

a processor configured to generate the plaintext message m by accessing a private key $\mathsf{usk} = \{r_{\mathsf{id}}\}$ for a user with an identity id by following steps:

if ${r_{\mathsf{id}}}^2 \equiv R_{\mathsf{id}} \pmod N$, setting $\nu = \varepsilon$, $\gamma = c$ and $\Delta = R_{\mathsf{id}}$; otherwise setting $\nu = \bar{\varepsilon}$, $\gamma = \bar{c}$ and $\Delta = uR_{\mathsf{id}}$;

determining $\sigma = J_N(\gamma^2 - 4\Delta)$,

setting

$$\tau = \begin{cases} J_N(\gamma + 2r_{\mathsf{id}}) & \text{if } \sigma = 1 \\ J_N((\gamma + 2r_{\mathsf{id}})(d - 2r_{\mathsf{id}})(d - \gamma)) & \text{if } \sigma = -1 \end{cases}, \text{ where } d = \mathcal{G}_{\mathsf{id}}(s); \text{ and}$$

determining the plaintext m as

$$m = \frac{1 - \nu \cdot \tau}{2}.$$

31. The apparatus of claim 30, wherein the private key $\mathsf{usk} = \{r_{\mathsf{id}}\}$ for the user with the identity id is generated by a master secrete key msk, wherein $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$, and if $R_{\mathsf{id}} \in \mathbb{QR}_N$ then $r_{\mathsf{id}} = {R_{\mathsf{id}}}^{1/2} \bmod N$; otherwise, $r_{\mathsf{id}} = (uR_{\mathsf{id}})^{1/2} \bmod N$.

32. The apparatus of claim 31, wherein $N = pq$, p and q are prime.

33. The apparatus of claim 32, wherein the master secret key is $\mathsf{msk} = \{p, q\}$.

34. The apparatus of claim 30, wherein the message $m \in \{0,1\}$.

35. The apparatus of claim 30, wherein d is the smallest nonnegative integer d such that such that $J_N(d^2 - 4\mathcal{H}(\mathsf{id})) = J_N(d^2 - 4u\mathcal{H}(\mathsf{id})) = -1$.

36. The apparatus of claim 30, wherein d is fixed as a global value to be used with all identities.

37. The apparatus of claim 30, wherein N is such that $J_N(-1) = -1$.

38. The apparatus of claim 37, wherein $d = 0$.

39. An apparatus for generating keys for an identity-based cryptosystem wherein a private key is used to decrypt ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ of a plaintext message m for a user with an identity id where

$$\varepsilon = (-1)^m J_N(t), \quad c^{(0)} = t + \frac{R_{id}}{t} \bmod N, \quad c^{(1)} = \frac{c^{(0)}d + 4R_{id}}{c^{(0)} + d} \bmod N,$$

$$\bar{\varepsilon} = (-1)^m J_N(\bar{t}), \quad \bar{c}^{(0)} = \bar{t} + \frac{uR_{id}}{\bar{t}} \bmod N, \quad \bar{c}^{(1)} = \frac{\bar{c}^{(0)}d + 4uR_{id}}{\bar{c}^{(0)} + d} \bmod N,$$

where N is a composite integer, $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$, $t, \bar{t} \in (\mathbb{Z}/N\mathbb{Z})^{\times}$, $d = \mathcal{G}_{\mathsf{id}}(s)$, $c = c^{(\beta_1)}$ and $\bar{c} = \bar{c}^{(\beta_2)}$ for random bits $\beta_1, \beta_2 \in \{0,1\}$, comprising:

a processor configured to set a master secret key msk, and generate, using the master secret key msk, the private key $\mathsf{usk} = \{r_{\mathsf{id}}\}$ for a user with an identity id, using public system parameters $\mathsf{mpk} = \{N, u, \mathcal{H}, \{\mathcal{G}_{\mathsf{id}}\}, s\}$, where N is a composite integer, $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$, s is bit-string, $\mathcal{H}$ is a cryptographic hash function mapping bit-strings to elements of $\mathbb{J}_N$ and $\{\mathcal{G}_{\mathsf{id}}\}$ is a family of functions mapping bit-strings to elements of a subset of $\mathbb{Z}/N\mathbb{Z}$ namely $\mathcal{G}_{\mathsf{id}}: \{0,1\}^* \to \mathbb{Z}/N\mathbb{Z}, s \mapsto d = \mathcal{G}_{\mathsf{id}}(s)$ such that $J_N(d^2 - 4\mathcal{H}(\mathsf{id})) = J_N(d^2 - 4u\mathcal{H}(\mathsf{id})) = -1$; and

a communication interface, coupled to a communication channel, configured to transmit the private key $\mathsf{usk} = \{r_{\mathsf{id}}\}$ via the communication channel.

40. The apparatus of claim 39, wherein $N = pq$, p and q are prime.

41. The apparatus of claim 40, wherein $\mathsf{msk} = \{p, q\}$.

42. The apparatus of claim 39, wherein N is such that $J_N(-1) = -1$.

43. A computer program product stored in non-transitory computer-readable storage media comprising computer-executable instructions for:

accessing the message m;

accessing a user identity id and public system parameters $\mathsf{mpk} = \{N, u, \mathcal{H}, \{\mathcal{G}_{\mathsf{id}}\}, s\}$, where N is a composite integer, $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$, s is a bit-string, $\mathcal{H}$ is a cryptographic hash function mapping bit-strings to elements of $\mathbb{J}_N$ and $\{\mathcal{G}_{\mathsf{id}}\}$ is a family of functions mapping bit-strings to elements of a subset of $\mathbb{Z}/N\mathbb{Z}$, namely $\mathcal{G}_{\mathsf{id}}: \{0,1\}^* \to \mathbb{Z}/N\mathbb{Z}, s \mapsto d = \mathcal{G}_{\mathsf{id}}(s)$ such that $J_N(d^2 - 4\mathcal{H}(\mathsf{id})) = J_N(d^2 - 4u\mathcal{H}(\mathsf{id})) = -1$;

generating a ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ of the message m using the user identity id, where

$$\varepsilon = (-1)^m J_N(t), \quad c^{(0)} = t + \frac{R_{id}}{t} \bmod N, \quad c^{(1)} = \frac{c^{(0)} d + 4R_{id}}{c^{(0)} + d} \bmod N,$$

$$\bar{\varepsilon} = (-1)^m J_N(\bar{t}), \quad \bar{c}^{(0)} = \bar{t} + \frac{uR_{id}}{\bar{t}} \bmod N, \quad \bar{c}^{(1)} = \frac{\bar{c}^{(0)} d + 4uR_{id}}{\bar{c}^{(0)} + d} \bmod N,$$

where $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$, $t, \bar{t} \in (\mathbb{Z}/N\mathbb{Z})^\times$, $d = \mathcal{G}_{\mathsf{id}}(s)$, choosing random bits $\beta_1, \beta_2 \in \{0,1\}$ and setting $c = c^{(\beta_1)}$ and $\bar{c} = \bar{c}^{(\beta_2)}$; and

transmitting the ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ via a communication channel.

44. A computer program product stored in non-transitory computer-readable storage media comprising computer-executable instructions for:

receiving the ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ for a user with an identity id via a communication channel; and

accessing the ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ of a plaintext message m, where

$$\varepsilon = (-1)^m J_N(t), \quad c^{(0)} = t + \frac{R_{id}}{t} \bmod N, \quad c^{(1)} = \frac{c^{(0)}d + 4R_{id}}{c^{(0)} + d} \bmod N,$$

$$\bar{\varepsilon} = (-1)^m J_N(\bar{t}), \quad \bar{c}^{(0)} = \bar{t} + \frac{uR_{id}}{\bar{t}} \bmod N, \quad \bar{c}^{(1)} = \frac{\bar{c}^{(0)}d + 4uR_{id}}{\bar{c}^{(0)} + d} \bmod N,$$

where N is a composite integer, $R_{\mathrm{id}} = \mathcal{H}(\mathrm{id})$, $t, \bar{t} \in (\mathbb{Z}/N\mathbb{Z})^{\times}$, $d = \mathcal{G}_{\mathrm{id}}(s)$, $c = c^{(\beta_1)}$ and $\bar{c} = \bar{c}^{(\beta_2)}$ for random bits $\beta_1, \beta_2 \in \{0,1\}$, wherein public system parameters $\mathrm{mpk} = \{N, u, \mathcal{H}, \{\mathcal{G}_{\mathrm{id}}\}, s\}$, where N is a composite integer, $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$, s is bit-string, $\mathcal{H}$ is a cryptographic hash function mapping bit-strings to elements of $\mathbb{J}_N$ and $\{\mathcal{G}_{\mathrm{id}}\}$ is a family of functions mapping bit-strings to elements of a subset of $\mathbb{Z}/N\mathbb{Z}$, namely $\mathcal{G}_{\mathrm{id}}: \{0,1\}^* \to \mathbb{Z}/N\mathbb{Z}, s \mapsto d = \mathcal{G}_{\mathrm{id}}(s)$ such that $J_N(d^2 - 4\mathcal{H}(\mathrm{id})) = J_N(d^2 - 4u\mathcal{H}(\mathrm{id})) = -1$; and

generating the message m by accessing a private key $\mathrm{usk} = \{r_{\mathrm{id}}\}$ for a user with an identity id by following steps:

if ${r_{\mathrm{id}}}^2 \equiv R_{\mathrm{id}} \pmod N$, setting $\nu = \varepsilon$, $\gamma = c$ and $\Delta = R_{\mathrm{id}}$; otherwise setting $\nu = \bar{\varepsilon}$, $\gamma = \bar{c}$ and $\Delta = uR_{\mathrm{id}}$;

determining $\sigma = J_N(\gamma^2 - 4\Delta)$,

setting

$$\tau = \begin{cases} J_N(\gamma + 2r_{\mathrm{id}}) & \text{if } \sigma = 1 \\ J_N((\gamma + 2r_{\mathrm{id}})(d - 2r_{\mathrm{id}})(d - \gamma)) & \text{if } \sigma = -1 \end{cases}, \text{ where } d = \mathcal{G}_{\mathrm{id}}(s); \text{ and}$$

determining the message m as

$$m = \frac{1 - \nu \cdot \tau}{2}.$$

45. A computer program product stored in non-transitory computer-readable storage media for generating keys for an identity-based cryptosystem wherein a private key is used to decrypt ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ of a plaintext message m for a user with an identity id where

$$\varepsilon = (-1)^m J_N(t), \quad c^{(0)} = t + \frac{R_{id}}{t} \bmod N, \quad c^{(1)} = \frac{c^{(0)}d + 4R_{id}}{c^{(0)} + d} \bmod N,$$

$$\bar{\varepsilon} = (-1)^m J_N(\bar{t}), \quad \bar{c}^{(0)} = \bar{t} + \frac{uR_{id}}{\bar{t}} \bmod N, \quad \bar{c}^{(1)} = \frac{\bar{c}^{(0)}d + 4uR_{id}}{\bar{c}^{(0)} + d} \bmod N,$$

where N is a composite integer, $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$, $t, \bar{t} \in (\mathbb{Z}/N\mathbb{Z})^{\times}$, $d = \mathcal{G}_{\mathsf{id}}(s)$, $c = c^{(\beta_1)}$ and $\bar{c} = \bar{c}^{(\beta_2)}$ for random bits $\beta_1, \beta_2 \in \{0,1\}$, comprising computer-executable instructions for:

setting a master secret key msk; and

generating, using the master secret key msk, the private key $\mathsf{usk} = \{r_{\mathsf{id}}\}$ for a user with an identity id, using public system parameters $\mathsf{mpk} = \{N, u, \mathcal{H}, \{\mathcal{G}_{\mathsf{id}}\}, s\}$, where N is a composite integer, $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$, s is bit-string, $\mathcal{H}$ is a cryptographic hash function mapping bit-strings to elements of $\mathbb{J}_N$ and $\{\mathcal{G}_{\mathsf{id}}\}$ is a family of functions mapping bit-strings to elements of a subset of $\mathbb{Z}/N\mathbb{Z}$ namely $\mathcal{G}_{\mathsf{id}}: \{0,1\}^* \to \mathbb{Z}/N\mathbb{Z}, s \mapsto d = \mathcal{G}_{\mathsf{id}}(s)$ such that $J_N(d^2 - 4\mathcal{H}(\mathsf{id})) = J_N(d^2 - 4u\mathcal{H}(\mathsf{id})) = -1$.

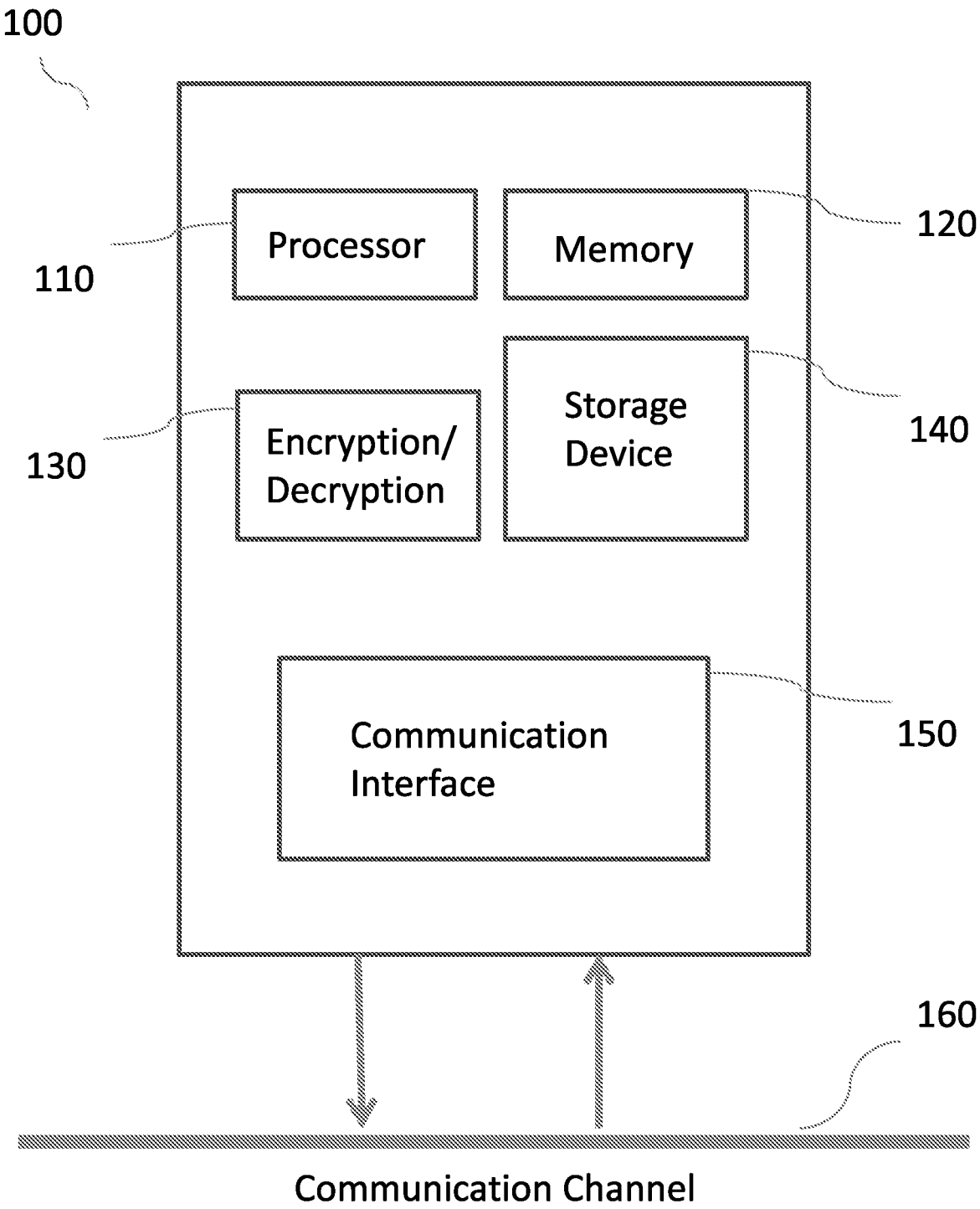
100
Processor
Memory
120
110
Encryption/
Decryption
Storage
Device
140
130
Communication
Interface
150
160
Communication Channel

Figure 1

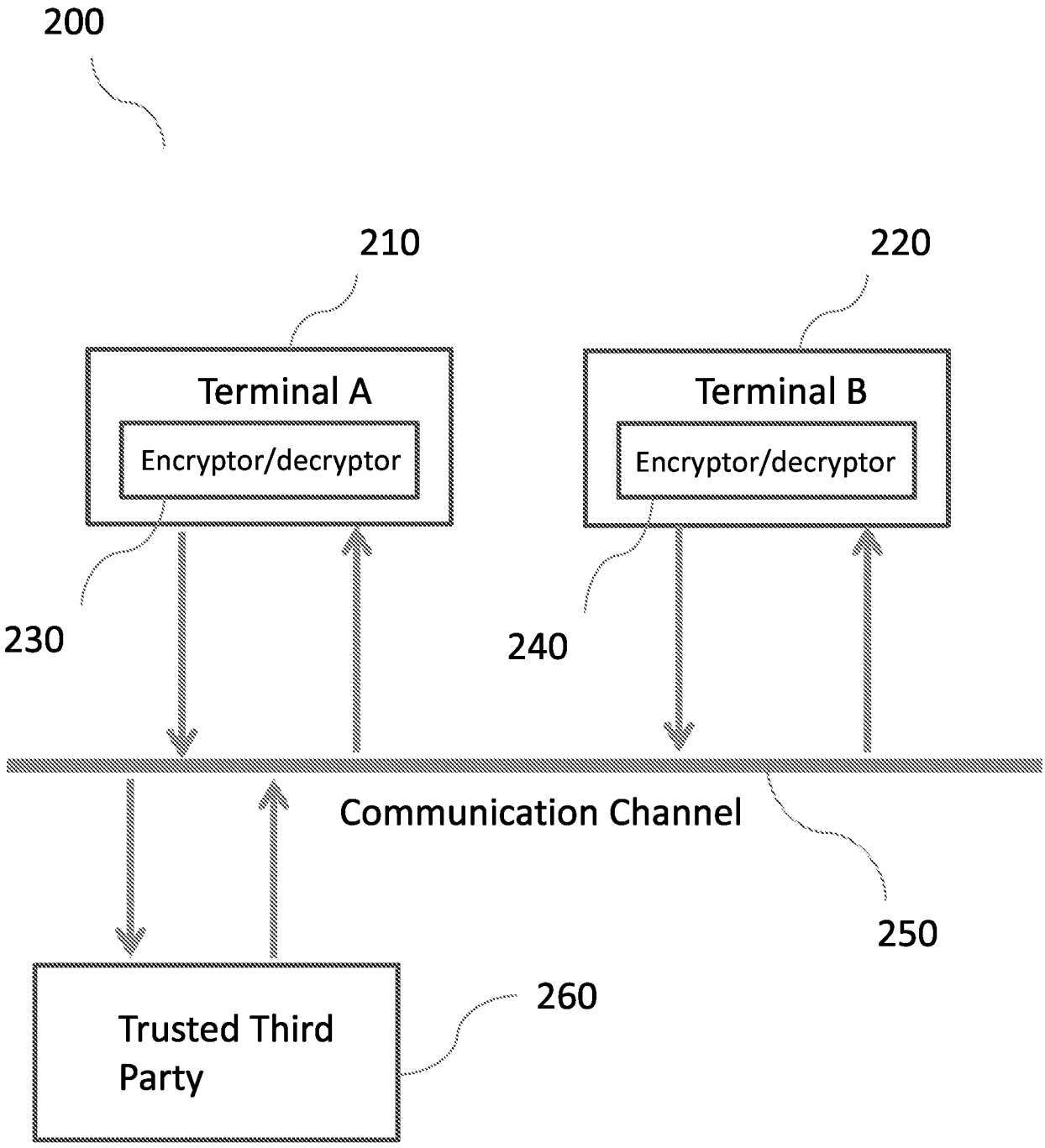
200
210
220
Terminal A
Encryptor/decryptor
Terminal B
Encryptor/decryptor
230
240
Communication Channel
250
Trusted Third Party
260

Figure 2

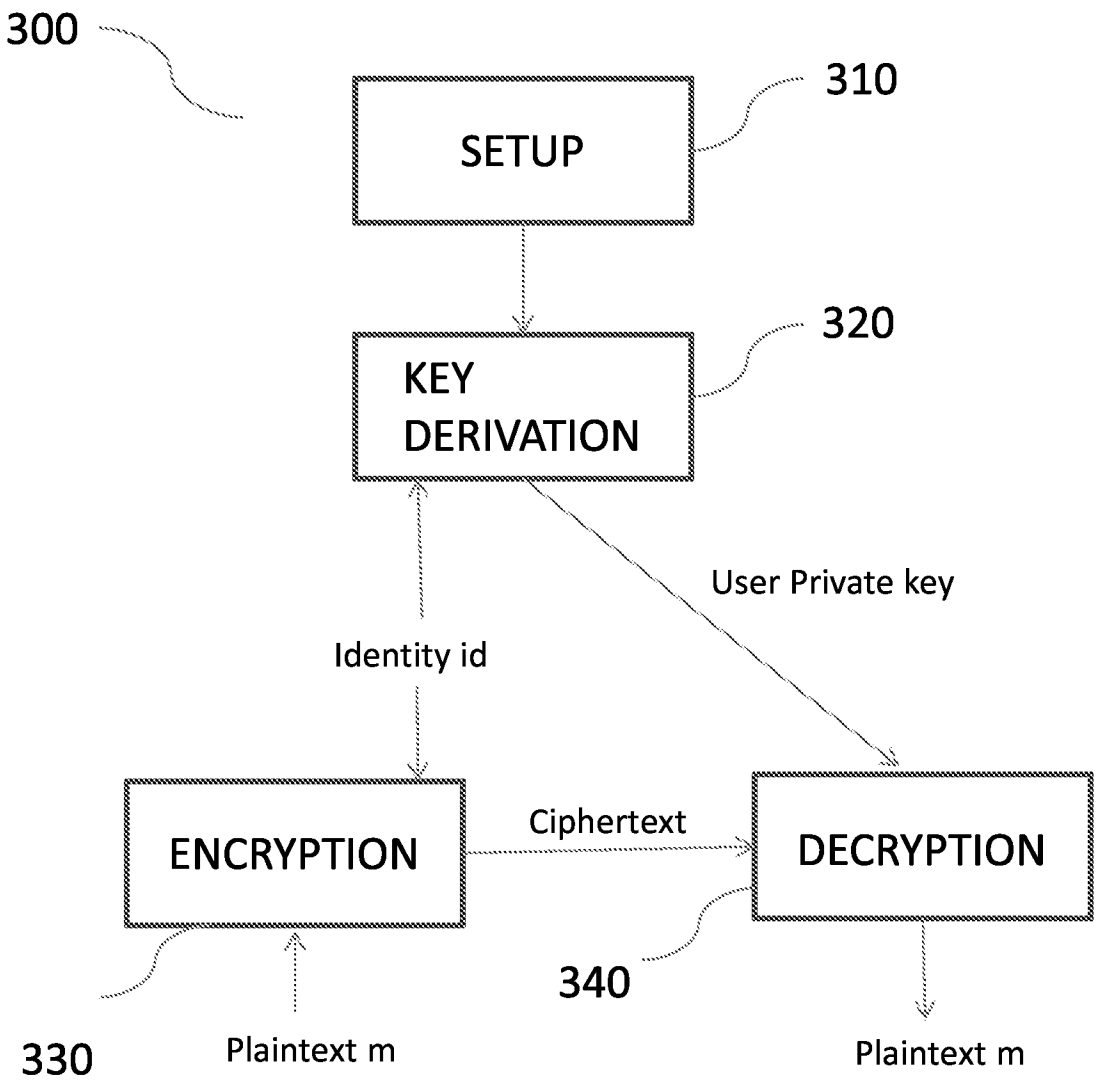
300
SETUP
310
KEY DERIVATION
320
User Private key
Identity id
ENCRYPTION
Ciphertext
DECRYPTION
340
330
Plaintext m
Plaintext m

Figure 3

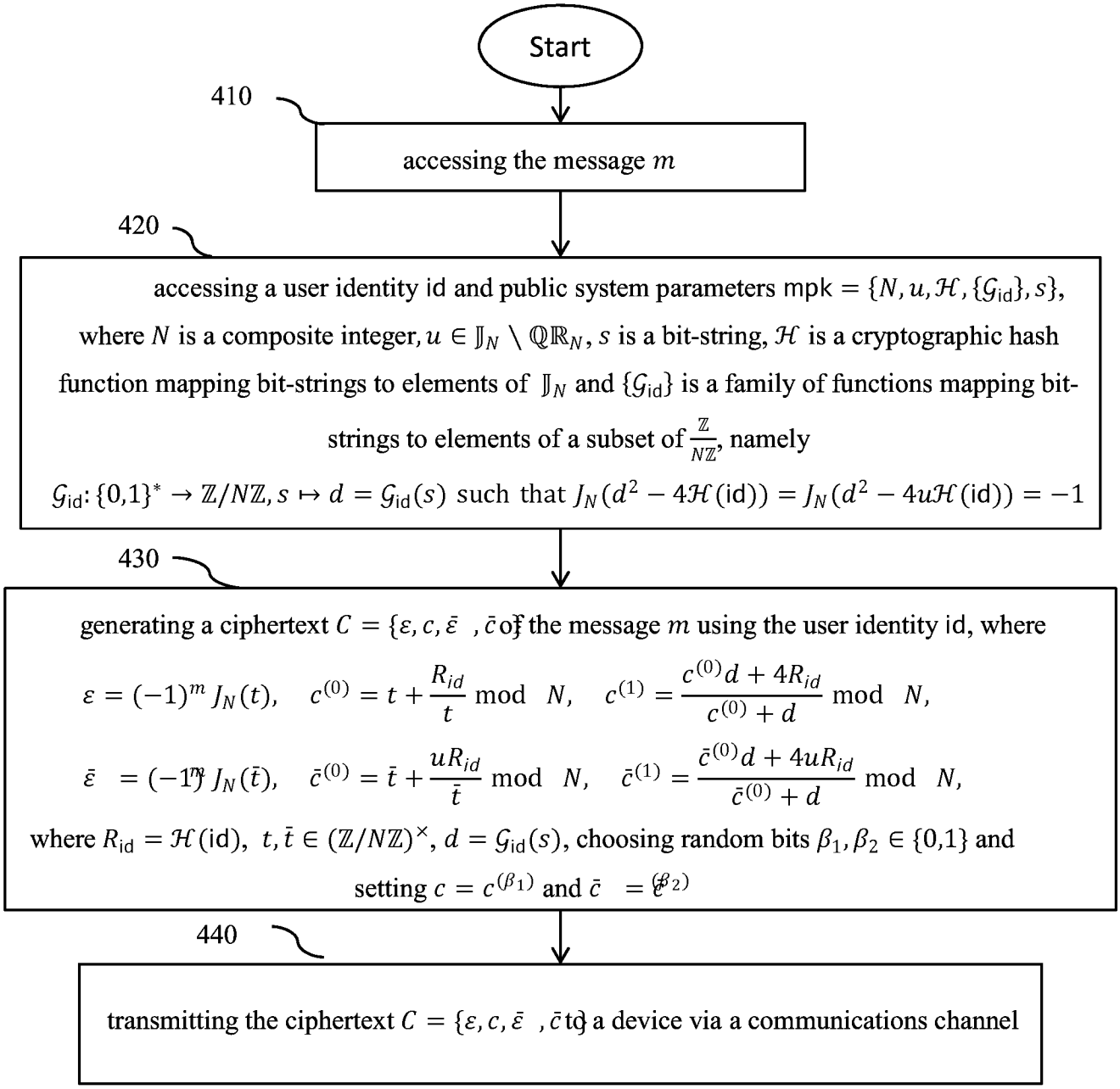
Start
410
accessing the message m
420
accessing a user identity id and public system parameters mpk $= \{N, u, \mathcal{H}, \{\mathcal{G}_{\mathrm{id}}\}, s\}$, where N is a composite integer, $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$, s is a bit-string, $\mathcal{H}$ is a cryptographic hash function mapping bit-strings to elements of $\mathbb{J}_N$ and $\{\mathcal{G}_{\mathrm{id}}\}$ is a family of functions mapping bit-strings to elements of a subset of $\frac{\mathbb{Z}}{N\mathbb{Z}}$, namely
$\mathcal{G}_{\mathrm{id}}: \{0,1\}^* \to \mathbb{Z}/N\mathbb{Z}, s \mapsto d = \mathcal{G}_{\mathrm{id}}(s)$ such that $J_N(d^2 - 4\mathcal{H}(\mathrm{id})) = J_N(d^2 - 4u\mathcal{H}(\mathrm{id})) = -1$
430
generating a ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ of the message m using the user identity id, where
$\varepsilon = (-1)^m J_N(t), \quad c^{(0)} = t + \frac{R_{id}}{t} \bmod N, \quad c^{(1)} = \frac{c^{(0)} d + 4R_{id}}{c^{(0)} + d} \bmod N,$
$\bar{\varepsilon} = (-1)^m J_N(\bar{t}), \quad \bar{c}^{(0)} = \bar{t} + \frac{uR_{id}}{\bar{t}} \bmod N, \quad \bar{c}^{(1)} = \frac{\bar{c}^{(0)} d + 4uR_{id}}{\bar{c}^{(0)} + d} \bmod N,$
where $R_{\mathrm{id}} = \mathcal{H}(\mathrm{id})$, $t, \bar{t} \in (\mathbb{Z}/N\mathbb{Z})^\times$, $d = \mathcal{G}_{\mathrm{id}}(s)$, choosing random bits $\beta_1, \beta_2 \in \{0,1\}$ and setting $c = c^{(\beta_1)}$ and $\bar{c} = \bar{c}^{(\beta_2)}$
440
transmitting the ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ to a device via a communications channel

Figure 4

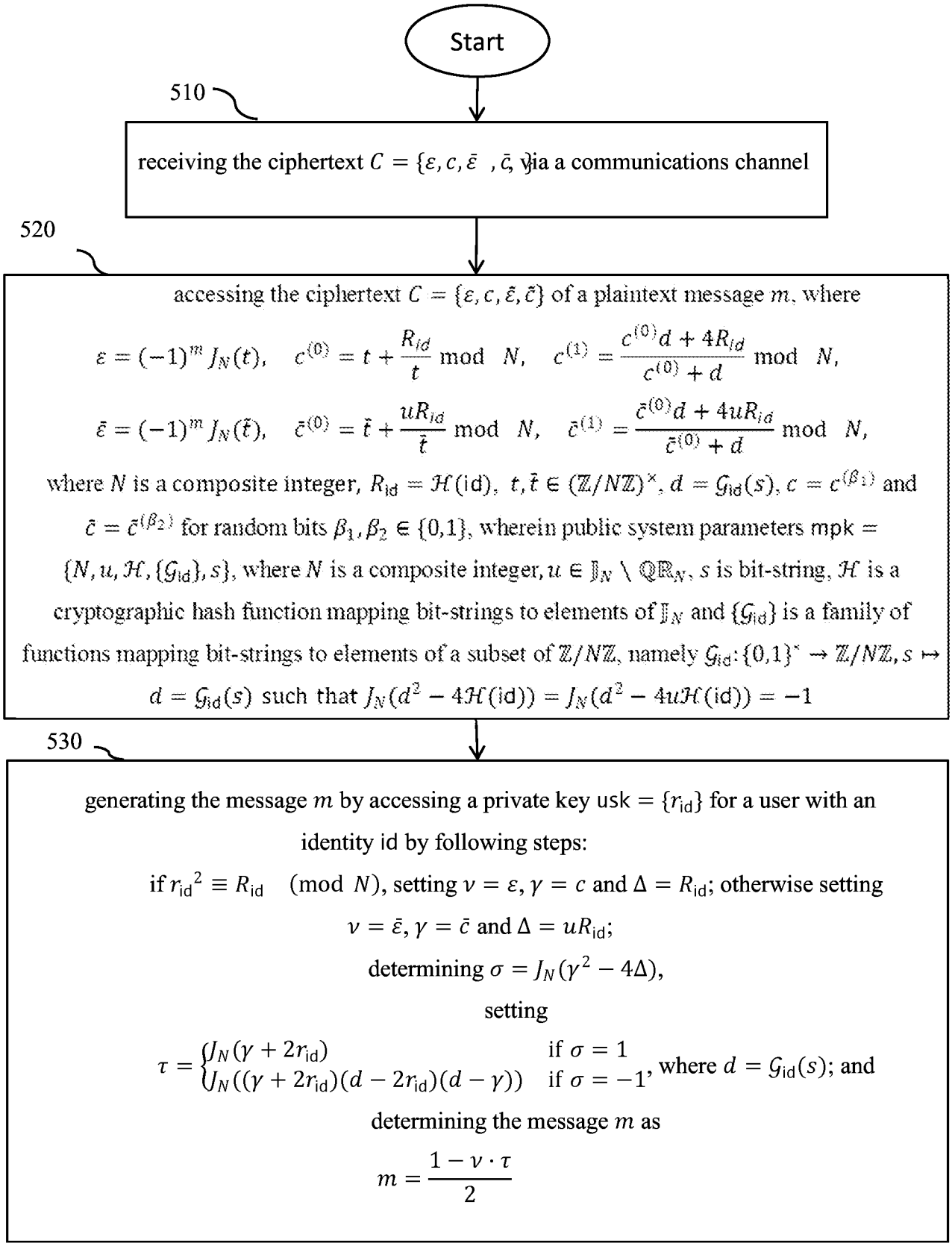
Start
510
receiving the ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ via a communications channel
520
accessing the ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ of a plaintext message m, where
$\varepsilon = (-1)^m J_N(t), \quad c^{(0)} = t + \frac{R_{id}}{t} \bmod N, \quad c^{(1)} = \frac{c^{(0)} d + 4R_{id}}{c^{(0)} + d} \bmod N,$
$\bar{\varepsilon} = (-1)^m J_N(\bar{t}), \quad \bar{c}^{(0)} = \bar{t} + \frac{uR_{id}}{\bar{t}} \bmod N, \quad \bar{c}^{(1)} = \frac{\bar{c}^{(0)} d + 4uR_{id}}{\bar{c}^{(0)} + d} \bmod N,$
where N is a composite integer, $R_{\mathrm{id}} = \mathcal{H}(\mathrm{id})$, $t, \bar{t} \in (\mathbb{Z}/N\mathbb{Z})^\times$, $d = \mathcal{G}_{\mathrm{id}}(s)$, $c = c^{(\beta_1)}$ and $\bar{c} = \bar{c}^{(\beta_2)}$ for random bits $\beta_1, \beta_2 \in \{0,1\}$, wherein public system parameters $\mathrm{mpk} = \{N, u, \mathcal{H}, \{\mathcal{G}_{\mathrm{id}}\}, s\}$, where N is a composite integer, $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$, s is bit-string, $\mathcal{H}$ is a cryptographic hash function mapping bit-strings to elements of $\mathbb{J}_N$ and $\{\mathcal{G}_{\mathrm{id}}\}$ is a family of functions mapping bit-strings to elements of a subset of $\mathbb{Z}/N\mathbb{Z}$, namely $\mathcal{G}_{\mathrm{id}}: \{0,1\}^* \to \mathbb{Z}/N\mathbb{Z}, s \mapsto d = \mathcal{G}_{\mathrm{id}}(s)$ such that $J_N(d^2 - 4\mathcal{H}(\mathrm{id})) = J_N(d^2 - 4u\mathcal{H}(\mathrm{id})) = -1$
530
generating the message m by accessing a private key $\mathrm{usk} = \{r_{\mathrm{id}}\}$ for a user with an identity id by following steps:
if $r_{\mathrm{id}}^2 \equiv R_{\mathrm{id}} \pmod N$, setting $\nu = \varepsilon, \gamma = c$ and $\Delta = R_{\mathrm{id}}$; otherwise setting $\nu = \bar{\varepsilon}, \gamma = \bar{c}$ and $\Delta = uR_{\mathrm{id}}$;
determining $\sigma = J_N(\gamma^2 - 4\Delta)$,
setting
$\tau = \begin{cases} J_N(\gamma + 2r_{\mathrm{id}}) & \text{if } \sigma = 1 \\ J_N((\gamma + 2r_{\mathrm{id}})(d - 2r_{\mathrm{id}})(d - \gamma)) & \text{if } \sigma = -1 \end{cases}$, where $d = \mathcal{G}_{\mathrm{id}}(s)$; and
determining the message m as
$m = \frac{1 - \nu \cdot \tau}{2}$

Figure 5

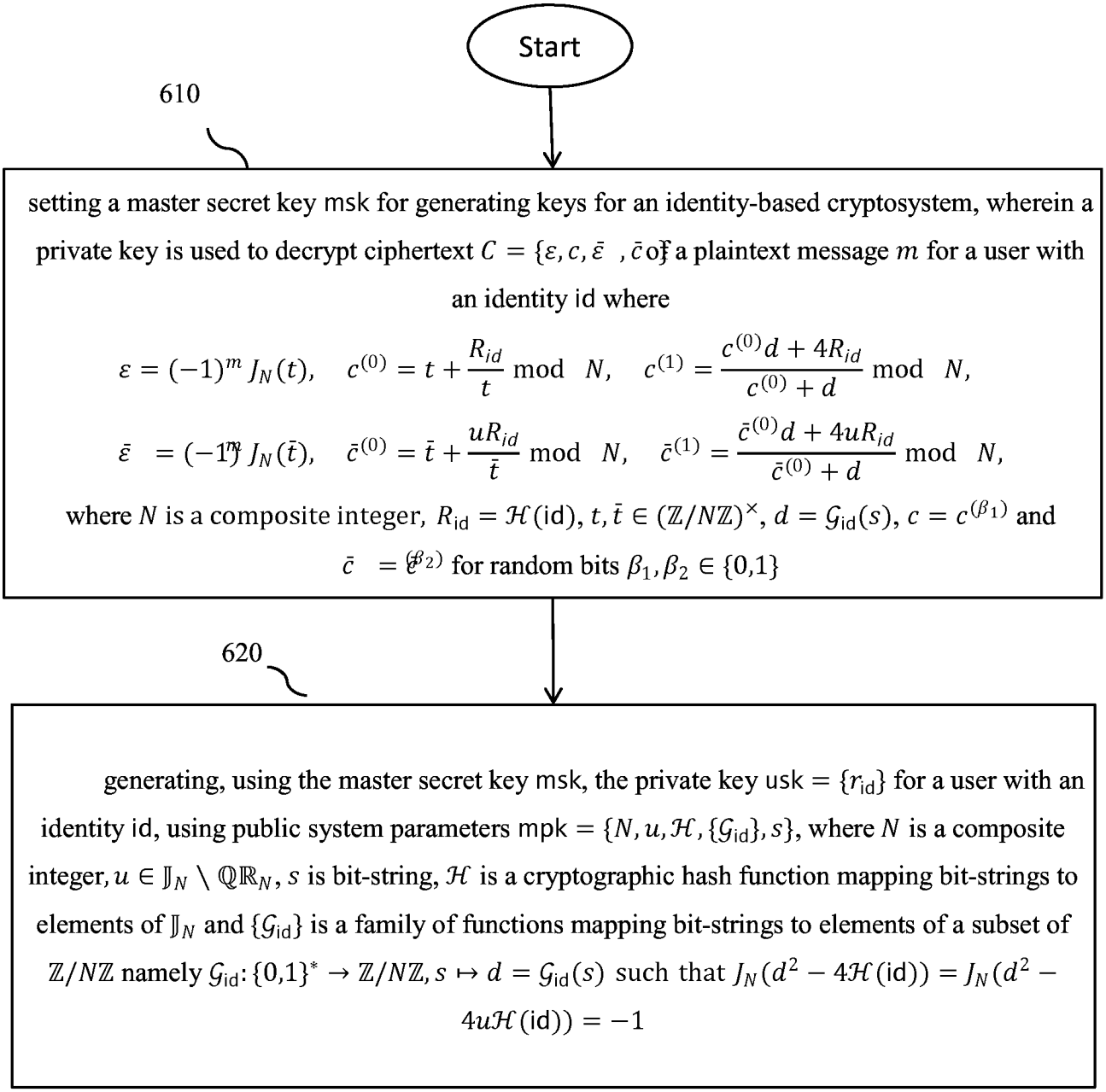
Start
610
setting a master secret key msk for generating keys for an identity-based cryptosystem, wherein a private key is used to decrypt ciphertext $C = \{\varepsilon, c, \bar{\varepsilon}, \bar{c}\}$ of a plaintext message m for a user with an identity id where
$\varepsilon = (-1)^m J_N(t), \quad c^{(0)} = t + \frac{R_{id}}{t} \bmod N, \quad c^{(1)} = \frac{c^{(0)}d + 4R_{id}}{c^{(0)} + d} \bmod N,$
$\bar{\varepsilon} = (-1)^m J_N(\bar{t}), \quad \bar{c}^{(0)} = \bar{t} + \frac{uR_{id}}{\bar{t}} \bmod N, \quad \bar{c}^{(1)} = \frac{\bar{c}^{(0)}d + 4uR_{id}}{\bar{c}^{(0)} + d} \bmod N,$
where N is a composite integer, $R_{\mathrm{id}} = \mathcal{H}(\mathrm{id}), t, \bar{t} \in (\mathbb{Z}/N\mathbb{Z})^{\times}, d = \mathcal{G}_{\mathrm{id}}(s), c = c^{(\beta_1)}$ and $\bar{c} = \bar{c}^{(\beta_2)}$ for random bits $\beta_1, \beta_2 \in \{0,1\}$
620
generating, using the master secret key msk, the private key $\mathrm{usk} = \{r_{\mathrm{id}}\}$ for a user with an identity id, using public system parameters $\mathrm{mpk} = \{N, u, \mathcal{H}, \{\mathcal{G}_{\mathrm{id}}\}, s\}$, where N is a composite integer, $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$, s is bit-string, $\mathcal{H}$ is a cryptographic hash function mapping bit-strings to elements of $\mathbb{J}_N$ and $\{\mathcal{G}_{\mathrm{id}}\}$ is a family of functions mapping bit-strings to elements of a subset of $\mathbb{Z}/N\mathbb{Z}$ namely $\mathcal{G}_{\mathrm{id}}: \{0,1\}^* \to \mathbb{Z}/N\mathbb{Z}, s \mapsto d = \mathcal{G}_{\mathrm{id}}(s)$ such that $J_N(d^2 - 4\mathcal{H}(\mathrm{id})) = J_N(d^2 - 4u\mathcal{H}(\mathrm{id})) = -1$

Figure 6

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2015/050670

A. CLASSIFICATION OF SUBJECT MATTER
INV. H04L9/08
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
EPO-Internal

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	GIUSEPPE ATENIESE ET AL: "Universally Anonymous IBE based on the Quadratic Residuosity Assumption", INTERNATIONAL ASSOCIATION FOR CRYPTOLOGIC RESEARCH,, vol. 20090616:200931, 11 June 2009 (2009-06-11), pages 1-17, XP061003455, [retrieved on 2009-06-11] cited in the application the whole document ----- -/--	1-45

[X] Further documents are listed in the continuation of Box C. [ ] See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
30 November 2015	09/12/2015

Name and mailing address of the ISA/	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Di Felice, M

1

Form PCT/ISA/210 (second sheet) (April 2005)

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	DAN BONEH ET AL: "Space-Efficient Identity Based Encryption Without Pairings", INTERNATIONAL ASSOCIATION FOR CRYPTOLOGIC RESEARCH,, vol. 20070520:125819, 10 May 2007 (2007-05-10), pages 1-38, XP061002317, [retrieved on 2007-05-10] cited in the application the whole document -----	1-45